Electromagnetic Ray-Tracing for the Investigation of Multipath and Vibration Signatures in Radar Imagery

Darren G. Muff

A dissertation submitted in partial fulfillment

of the requirements for the degree of

Doctor of Philosophy

of

University College London.

Department of Electronic Engineering

University College London

August 4, 2018

I, Darren G. Muff, confirm that the work presented in this thesis is my own. Where information has been derived from other sources, I confirm that this has been indicated in the work.

Abstract

SYNTHETIC Aperture Radar (SAR) imagery has been used extensively within UK Defence and Intelligence for many years. Despite this, the exploitation of SAR imagery is still challenging to the inexperienced imagery analyst as the non-literal image provided for exploitation requires careful consideration of the imaging geometry, the target being imaged and the physics of radar interactions with objects. It is therefore not surprising to note that in 2017 the most useful tool available to a radar imagery analyst is a contextual optical image of the same area.

This body of work presents a way to address this by adopting recent advances in radar signal processing and computational geometry to develop a SAR simulator called SARCASTIC (**SAR** Ray-**Cast**er for the **I**ntelligence **C**ommunity) that can rapidly render a scene with the precise collection geometry of an image being exploited. The work provides a detailed derivation of the simulator from first principals. It is then validated against a range of real-world SAR collection systems. The work shows that such a simulator can provide an analyst with the necessary tools to extract intelligence from a collection that is unavailable to a conventional imaging system.

The thesis then describes a new technique that allows a vibrating target to be detected within a SAR collection. The simulator is used to predict a unique scattering signature - described as a *one-sided paired echo*. Finally an experiment is described that was performed by Cranfield University to specifications determined by SARCASTIC which show that the unique radar signature can actually occur within a SAR collection.

Impact Statement

A key tool in the arsenal of the intelligence community has been the exploitation of synthetic aperture radar (SAR) imagery as it provides the ability to image in all weather conditions and throughout the day or night. A reduction in satellite launch costs, and an increase in availability of high frequency microwave components, has advanced the accessibility of space-based SAR in recent years. As a result, orbital SAR imagery is becoming increasingly utilised by bodies outside of the intelligence community such as news and media providers and the banking sector. Although the ability to launch and operate a SAR satellite has become more affordable, the exploitation of SAR imagery, requires a deep level of understanding of the nature of the imaging operation. The work described in this thesis provides several contributions that will increase this understanding. It provides a foundation for the scientific community by describing the nature of electromagnetic interaction and scattering from simulated computer aided design (CAD) models starting from Maxwell's equations and first principles. In developing a high fidelity simulator that operates in a useful timescale it is necessary to develop and describe modern ray-tracing techniques and in particular how ray-tracing is accelerated through the use of hierarchical data structures and graphical processing units (GPUs). A novel implementation that outperforms previously published work is likely to be useful to the computer games industry as well as those developing modern ray-tracing algorithms such as the fields of astronomy and medicine. The thesis concludes by presenting two approaches to the exploitation of SAR imagery. The first demonstrates how multi-path analysis can be performed to increase an analyst's understanding of the inside of a building. The second provides a technique to detect vibrating and mod-

ulating features within a SAR collection such as those from an operating generator. This may have an impact on the intelligence and defence communities as it potentially increases the amount of information that can be obtained of a target area or facility. Outside of defence, this additional level of scene characterisation will also contribute to humanitarian efforts such as disaster-relief operations which are increasingly using space-based SAR. Although the thesis is specifically related to SAR exploitation, the radar interactions are described in terms of individual pulses and may therefore have an impact on other radar exploitation and characterisation areas such as automotive radar. Finally, there may also be application to civil surveillance systems (for example those used by the police and security) interested in the simulation, characterisation and detection of helicopters and drones.

Contents

List of Figures **18**

List of Tables **19**

Symbols **21**

Acknowledgements **23**

Thesis Publications **25**

1 Introduction **27**

1.1 Novel Contributions 29

1.2 Thesis Structure 30

1.3 The Evolution of SAR Image Formation 32

1.3.1 Optical Processing Origins 32

1.3.2 Beyond Unfocused SAR 34

1.3.3 Digital Image Formation 36

1.4 Atypical Scattering in SAR Images 40

1.4.1 Target Motion 41

1.4.2 Changes in Target Radar Cross Section 43

1.4.3 Radar Multipath 43

1.4.4 Start-Stop Approximation 44

1.4.5 Varying Frequency Response 44

1.4.6 Inhomogeneous Propagation Medium 45

1.5 From Radar to Synthetic Aperture Radar 46

1.5.1 Introduction 46
1.5.2 Components of a Radar System 46
1.5.3 Single Pulse Physics Model 47
1.5.4 Synthetic Aperture Analysis 54
1.5.5 Extension to Spotlight Mode SAR Image Formation 57
1.5.6 Received Signal 62
1.6 Summary 64

2 Background to Electromagnetism and Physical Optics 65
2.1 Derivation of Scattered EM Fields at the Receiver 66
2.1.1 The Approach 67
2.1.2 Defining the Fields in Terms of their Vector Potentials 67
2.1.3 Solving the Vector Potentials 75
2.1.4 Simplification Using the Far-Field Approximation 83
2.1.5 Rebuilding the E and H Fields From the Vector Potentials . . 85
2.2 Electromagnetic Material Properties 90
2.2.1 Electrical and Magnetic Material Properties 93
2.2.2 Boundary Conditions 95
2.2.3 Combination Sheet 101
2.2.4 Translation into Facet Coordinates 105
2.2.5 Transmission and Reflection Coefficients 107
2.3 The Induced Surface Currents 120
2.3.1 Electric Surface Currents 122
2.3.2 Magnetic Surface Currents 125
2.3.3 Summary 126
2.4 Scattered Fields From a Planar Surface 127
2.4.1 Introduction 127
2.4.2 Parallel Incident Polarisation 128
2.4.3 Perpendicular Incident Polarisation 130
2.4.4 The E-Field for a Rectangular Facet 132
2.4.5 Parallel Polarisation Case 133

2.4.6 Perpendicular Polarisation Case 134
2.4.7 Arbitrary Incident Polarisation 136
2.4.8 The E-Field for a Triangular Facet 137
2.5 Summary 144

3 All About Triangles **147**
3.1 Introduction 147
3.2 Mesh or Soup? 148
3.3 Delaunay Triangulation 151
3.4 Material Properties 152
3.5 Ray-Triangle Intersection Optimisation 155
3.5.1 Generic Formulation of Ray-Triangle Intersection 156
3.5.2 Barycentric Coordinates 159
3.5.3 Simplification Through Projection 160
3.5.4 Implementation Efficiencies 163
3.6 Summary 168

4 Accelerated Data Structures for Efficient Ray Tracing **169**
4.1 Checking for Bounding Volume Intersection 170
4.2 Acceleration Structures for Scene Sub-Division 173
4.2.1 Bounding Volume Hierarchies (BVH) 174
4.2.2 Spatial Subdivision (Octrees, Binary Space Partitions, etc.) . 174
4.3 Building Efficient Kd-Trees and Building Them Efficiently 177
4.3.1 Overview of a Kd-Tree 177
4.3.2 Basic Kd-Tree Construction 179
4.3.3 The Surface Area Heuristic 180
4.3.4 Optimal Split Plane Selection 182
4.3.5 Automatic Termination Criteria 183
4.3.6 Triangle Classification 184
4.3.7 Kd-Tree Construction in *O(NlogN)* 187
4.3.8 GPU Optimisation 192

4.4 Performance Results of SARCASTIC's Kd-tree Builder 195
4.5 Summary 199

5 Implementation of SARCASTIC 201
5.1 Program Structure 201
5.2 Optimisation Using Pulse Under-Sampling 205
5.3 Results 206
5.4 Recommendations for Improvement 210
5.5 Summary 210

6 Multipath Analysis 213
6.1 Polarisation 214
6.1.1 Pauli Decomposition 215
6.2 Ray Unravelling 218
6.3 Application to Real Data 220
6.4 Summary 225

7 Motion And Vibration 229
7.1 Introduction 229
7.2 Form of Modulating Signal 230
7.2.1 The Effect of Polar Formatting the Received Signal 234
7.2.2 Other SAR Image Formation Approaches 237
7.2.3 Modulating Signatures 237
7.2.4 The Form of the Modulating Target in the SAR Image 240
7.3 Detecting Real-World Vibrating Signatures 243
7.3.1 Bright Spark SAR Data Collection System 243
7.3.2 Vibration Experiment 245
7.3.3 The SAR Multi-look Movie 249
7.4 Exploration of Unusual Effects 252
7.4.1 GB-SAR 252
7.4.2 Multipath Experiments 253
7.4.3 Vibration Experiments 254

7.5 Summary 257

8 Conclusions **259**

8.1 Summary 261

8.2 Recommendations for Future Work 261

Acronyms **265**

Glossary **269**

Bibliography **273**

Colophon **291**

List of Figures

1.1 Process flow for an exploitation tool that will simulate a radar scene using collected raw data and a CAD model 29
1.2 Operation of a SAR Sensor 33
1.3 SAR Optical Correlator 34
1.4 Comparison of Strip-Map and Spotlight Modes of SAR Operation . 35
1.5 SAR Amplitude Modulation 42
1.6 SAR Multipath in Action 44
1.7 Propagation Effects on SAR Imagery 45
1.8 Block Diagram of Key Components to a Radar Sensor. 47
1.9 Diagram of a Pulse Train Transmitted From a Radar 48
1.10 Complex Signal Reconstruction 50
1.11 Representation of a Chirp 50
1.12 Matched Filter Impulse Response 54
1.13 Geometry Description for Strip-Map SAR Operation 55
1.14 Range Ambiguities 58
1.15 Azimuth Deramp Process Used in Spotlight Mode Imaging 61

2.1 Different EM Approaches to Solving the EM Field Equation 67
2.2 Infinitesimal Point Charge 75
2.3 How to Solve Linear Homogeneous Partial Differential Equations . . 77
2.4 Case When Source is Located Away From the Origin 81
2.5 Validity of the Far-Field Approximation 83
2.6 Steps to Determine Surface Currents From an Incident EM Field . . 92
2.7 Resistance Vs Resistivity 93

2.8 Boundary of a Perfectly Conducting Medium 95
2.9 Electric Field Lines at the Boundary of a Perfect Electrical Conductor 97
2.10 Magnetic Field Boundary Conditions for Perfect Electrical Conductor 99
2.11 Sheet Resistance 102
2.12 Coordinate Transformation From Global Coordinates to the Facet . . 106
2.13 Coordinate System for a Ray Incident on a Reflecting Sheet 107
2.14 Polarisation is Defined Relative to the Plane of Incidence 108
2.15 Scattering Point Coordinate System 121
2.16 EM Ray Incident on a Planar Surface 132
2.17 Visualisation of Ludwig Integration 138
2.18 RCS of a Flat Plate 145

3.1 Comparison of a Triangle Soup to a Triangle Mesh 149
3.2 Delaunay Triangulation 152
3.3 Creating a Rough Surface Mesh 154
3.4 Application of MATERIALISE Algorithm 155
3.5 Derivation of a Ray Intersecting a Plane 157
3.6 Ceva's Theorum 159
3.7 Barycentric Coordinate Projection 161

4.1 Ray-Scene Interaction 170
4.2 The Advantage of a Binary Tree 173
4.3 Spatial Subdivision 174
4.4 Building a Kd-tree 178
4.5 The Cost of SAH Split Candidates 183
4.6 Plane Sweep Approach to Determine SAH Split Candidates 187
4.7 Triangle Splitting 191
4.8 The Stanford Dragon 198
4.9 The Stanford Bunny and Armadillo 198
4.10 Kd-tree Construction of a Main Battle Tank 199

5.1 Process Flow of a Typical Run of SARCASTIC 202

5.2 Bistatic RCS Measurements of Trihedral and Nighthawk 203
5.3 Simulated Trihedral Farm 207
5.4 CAD Model of T-72 Main Battle Tank 208
5.5 CAD Model of T-72 After MATERIALISE 208
5.6 Comparison of Real and Simulated Images of T-72 MBT 209

6.1 Silo CAD Model 213
6.2 SARCASTIC SAR Image of the Silo 214
6.3 Pauli Decomposition of the Silo CAD Model 216
6.4 Original Polarisation Responses 218
6.5 Process Flow to Ray-unravel a Signature Within a SARCASTIC Image 219
6.6 Three-Bounce Interaction and Pauli Decomposition 220
6.7 FSAR X-band SAR Image of Dstl Porton Down 221
6.8 Porton Down Fire Station 222
6.9 F-SAR Image of Porton Down Fire Station 223
6.10 Left Bay Within Fire Station 224
6.11 Comparison of Fire Station SAR Imagery 224
6.12 Ray-Tracing of Fire Station Interior 226

7.1 Amplitude Modulation 230
7.2 Simple Representation of SAR Collection Geometry 230
7.3 Target Range as a Function of Position Through Synthetic Aperture . 231
7.4 Pulse-to-Pulse Range Rate as a Function of Along Track Position . . 232
7.5 De-Ramped Signal History 233
7.6 The Effect of Polar Formatting 235
7.7 Range History of a Target as a Function of Synthetic Aperture Angle θ 235
7.8 Target Signal Vs Along Track Position 236
7.9 Phase of Modulating Target Vs Position 240
7.10 Cartoon of a Modulating Target 240
7.11 Five Sub-Looks of Modulating Target 241
7.12 Holographic Explanation of Modulating Target 242

7.13 Bright Spark Image of Bruntingthorpe Airfield 245
7.14 Bright Spark Image of Westcott 246
7.15 Image of Generator in Westcott Bright Spark Scene 247
7.16 Sketchup® Model of Generator 247
7.17 Comparison of Generator Images 247
7.18 Dominant Scattering on Generator 248
7.19 SARCASTIC Image of Vibrating Generator 249
7.20 Individual *Looks* of the SARCASTIC Simulated Generator 250
7.21 SAR Multi-Look Movie of Vibrating Generator 250
7.22 Comparison of Multi-Look Movie 251
7.23 Cranfield University's GB-SAR 252
7.24 GB-SAR Multipath of Oil Drums 253
7.25 Comparison of Multipath Signatures From Barrels 254
7.26 Controlled Vibration Platform (CVP) 255
7.27 SARCASTIC Simulation Through a Wall 255
7.28 GB-SAR Image of Controlled Vibration Platform 256
7.29 One-Sided Paired Echoes 257

List of Tables

1.1 Phase Terms Present in the Received SAR Signal 64

2.1 Definition of Parameters in Maxwell's Equations 68
2.2 Conductivity of Selected Materials 95

3.1 Material Properties Used in SARCASTIC 153
3.2 Clock Cycles for Ray-Triangle Intersection 158
3.3 CPU Cycles When Using Barycentric Coordinates 163
3.4 Clock Cycles Using Barycentric Coordinates and Pre-Factoring . . . 165

4.1 Modified *OutCodes* for a 3D Cohen-Sutherland Clipping Algorithm 171
4.2 Comparison of Performance of FASTKDTREE 197

6.1 Nominal F-SAR Parameters for Hydravision II Trial 221

7.1 Nominal Operating Parameters for Bright Spark SAR System 244
7.2 Combined Field Contributions on Generator 248
7.3 Calculation of the Vibration Frequency of the Generator 252

Symbols

Symbol	Description	SI Unit
A	Magnetic vector potential.	$V\,s\,m^{-1}$
$A(t)$	Amplitude of a signal as a function of time.	V
B	Bandwidth.	Hz
$\vec{B}$	Magnetic flux density.	T or $N\,(mA)^{-1}$
c	Velocity of light in vacuum.	$299\,792\,458 m\,s^{-1}$
$\vec{D}$	Electric flux density.	$C\,m^{-2}$
$\vec{E}$	Electric field.	V
F	Electric vector potential F.	$A\,s\,m^{-1}$
f_c	Centre frequency.	Hz
f_d	Doppler frequency shift.	Hz
g	Grazing angle. The angle subtended at the target between the horizontal and the sensor.	rad
$\vec{H}$	Magnetic field.	$A\,m^{-1}$
$\vec{J}$	Electric current density.	$A\,m^{-2}$
j	$\sqrt{-1}$.	
L_{sa}	Synthetic aperture length.	m
$\vec{M}$	Magnetic current density	
$O(N)$	Big O notation used in computer science.	
R	Range	m
R_m	Sheet magnetic conductivity.	$S\,\square^{-1}$
R_s	Sheet resistivity.	$\Omega\,\square^{-1}$
T	Interpulse period.	s

Symbol	Description	SI Unit
t	Time.	s
$T_{\parallel}$	Parallel transmission coefficient.	
$T_{\perp}$	Perpendicular transmission coefficient.	
T_{sa}	Synthetic aperture duration.	s
T_p	Length of a radar pulse.	s
v	Velocity.	$m\,s^{-1}$
ω	Angular frequency.	$rad\,s^{-1}$
Z_0	Impedance of free space.	376.7303Ω
δ_r	Range resolution.	m
δ_x	Along track (azimuth) resolution.	m
ε	Permittivity (dielectric constant).	$F\,m^{-1}$
ε_0	Permittivity of free space.	$8.854\times10^{-12}Fm^{-1}$
ε_r	Relative permittivity.	≥ 1
$\Gamma_{\parallel}$	Parallel reflection coefficient.	
$\Gamma_{\perp}$	Perpendicular reflection coefficient.	
γ	Chirp rate.	$Hz\,s^{-1}$
λ_c	Centre wavelength corresponding to f_c.	m
μ	Permeability.	$H\,m^{-1}$ or $N\,A^2$
μ_0	Magnetic constant.	$4\pi\times10^{-7}\,H\,m^{-1}$
μ_r	Relative permeability.	
ϕ	Azimuth angle around $\vec{z}$ axis in plane of facet.	rad
ϕ	Measurement of phase.	rad
π	A mathematical constant whose value is the ratio of any circle's circumference to its diameter.	
ρ_e	Electric charge density.	$C\,m^{-3}$
ρ_m	Magnetic charge density.	
σ	Electric charge density.	$S\,m^{-1}$
σ_t	Radar cross section of a target.	m^2
θ	Incidence angle in plane of facet.	rad

Acknowledgements

This work would not have been completed without a lot of help from friends, family and colleagues.

It was funded jointly by the Defence Science and Technology Laboratory and Joint Forces Intelligence Group (and its predecessor JARIC - the Joint Air Reconnaissance Intelligence Centre). I would therefore like to thank both command teams for their continued funding and support, in particular Roly, Miles and Penny.

Sincere thanks also to my supervisor, Professor Hugh Griffiths.

There are a number of friends and colleagues amongst the 'five-eyes' intelligence community that I am deeply indebted to, particularly the amazing SAR pioneers at the NRO but also NRL and NGA (USA), DRDC (Canada), and DST Group (Australia). It would be ill advised to name individuals but you know who you are and how you have made the world safer.

Particular thanks are owed to my dear departed friend Charles 'Jack' Jakowatz and the team from Sandia National Labs who provided experimental data to work with (thanks Ireena and Paul). Also Dan and Brandon from Cranfield University for their GB-SAR data. Finally Malcolm from Thales, UK for the Bright Spark Data.

I owe a significant debt of gratitude to the DSTL team at JFIG, especially Matt, Claire, Dave, Natalie and Mark.

I'd like to thank my two sons Jake and Jed for allowing me to take time off from parenting to pursue various science and engineering projects, rarely finishing them (and when I did the wheels usually fell off). Finally, and most importantly, thanks to my wife Joanne who, as well as being an amazing wife and mother, picked up a range of DIY and fatherly duties so that I could immerse myself in cyberspace.

Thesis Publications

D. André, D. Blacknell, D. G. Muff, and M.R. Nottingham. The physics of vibrating scatterers in SAR imagery. In *Proc. of SPIE - The International Society for Optical Engineering*, volume 8051, 2011.

D. André, D. Blacknell, D.G. Muff, and M. Nottingham. The physics of vibrating scatterers in SAR imagery. In Edmund G. Zelino and Frederick D. Garber, editors, *Aperture, Orlando, Florida, United States. SPIE*, 2011.

B. Corbett, D. André, D.G. Muff, and M. Finnis. Imaging SAR phenomenology of concealed vibrating targets. In *12th European Conference on Synthetic Aperture Radar, EUSAR 2018, Aachen, Germany*, 4 June 2018.

D.G. Muff, D. André, B. Corbett, M. Finnis, D. Blacknell, M.R. Nottingham, C. Stevenson, and H. Griffiths. Comparison of vibration and multipath signatures from simulated and real SAR images. In *IET Radar*, Belfast, UK, 2017.

D.G. Muff, M. Nottingham, D. André, B. Barber, D. Blacknell, and H. Griffiths. Characterisation of modulating signatures within sar imagery. In *International Conference on Synthetic Aperture Sonar and Synthetic Aperture Radar*, volume 32 of *Proceedings of the Institute of Acoustics*, pages 104–109. Curran Associates, Inc. (Nov 2010), 2010.

D.G. Muff, M. Stevens, D. Blacknell, R. Nottingham, M. Stevenson, and H. Griffiths. Detecting vibrating targets in fine resolution SAR imagery. In *12th European Conference on Synthetic Aperture Radar, EUSAR 2018, Aachen, Germany*, 4 June 2018.

Chapter 1

Introduction

The exploitation of synthetic aperture radar (SAR) data has become increasingly important in the field of Earth Observation (EO), particularly within the military and intelligence communities where rapid tactical imagery is provided through a SAR's ability to see in all weather and at daytime or night time. Another advantage of SAR for Earth Observation, where the range from sensor to target area is large, is provided by the radar's ability to provide fine resolution imagery regardless of stand-off range. This is due to the SAR image resolution being determined by more manageable engineering constraints such as transmit bandwidth and dwell times rather than the diameter of the mirror in an optical system.

SAR imagery has therefore been used extensively within Defence and Intelligence for many years. Despite this, the exploitation of SAR imagery is still challenging to the inexperienced imagery analyst as the non-literal image provided for exploitation requires careful consideration of the imaging geometry, the target being imaged and the physics of radar interactions with objects. It is therefore not surprising to note that in 2017 the most useful tool available to a radar imagery analyst is a contextual optical image of the same area.

The increase in number of imaging satellites has also had the effect of highlighting the discipline of Earth Observation to the UK's global adversaries. It is now commonly recognised that satellites regularly image the globe. As a result many illicit activities that would routinely take place in the open (such as illegal drug production or terrorist training facilities) have now moved out of direct line of sight

into buildings or underground facilities.

To try and address this it was decided that a new exploitation tool would be developed that would allow an analyst to better interrogate and explore the nuances of SAR imagery in order to understand subtle tell-tale signatures in a more forensic way. The approach explored here in this work is to use modern computer science techniques to rapidly simulate a target area or building within a SAR image. If the simulation can be driven by a computer aided design (CAD) model that can be easily adjusted, and the simulated radar image rendered quickly enough to provide a useful comparison, then an analyst can start to explore their SAR image by *hypothesis testing* using an analytical understanding of the target area and inserting objects and features into the CAD model to provide clues as to the true scene's content.

Obviously this approach is limited by the accuracy of the CAD model being used and unknown errors in the SAR collection and processing chain. The first problem is more easily addressed by constraining the target area to man-made objects that do not change significantly over time such as buildings. Analysts usually look at such areas many times and have a detailed understanding of the size, shape and orientation of these targets. The second source of error though can be addressed by integrating the simulation into the processing chain. In doing this, precise information regarding the transmit and receive locations and times of each radar pulse can be used to simulate the scene. If the raw *phase history data* is then simulated and passed through the same image formation processor (IFP) used to form the true SAR image then any processor-induced artifacts can be removed (or at least made the same in both the true scene and the simulation).

A flow chart describing this approach and the operation of such an exploitation tool can be seen in Figure 1.1.

This thesis describes the physical processes undertaken by a synthetic aperture radar in terms of the collection of electromagnetic fields, and the digitisation and processing of the measured voltages into a final SAR image. It describes the construction of a new SAR simulator called SARCASTIC (SAR Ray-Caster for the Intelligence Community). The output of the simulator is compared to two different

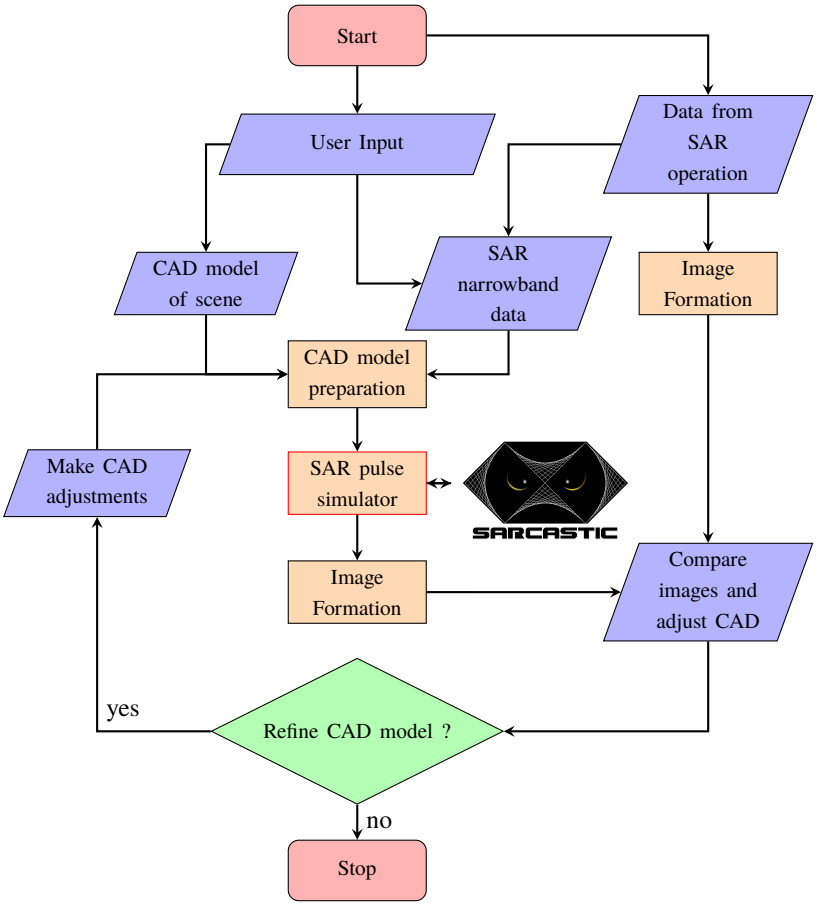


Figure 1.1: Process flow for an exploitation tool that will simulate a radar scene using collected raw data and a CAD model

sets of radar data. Two different types of radar phenomenology are explored using the simulator - *multipath* and vibration signatures. These are subtle effects caused by mismatches between true radar scattering and assumptions made within a SAR IFP and therefore ideally suited for a comparison between a simulator and real-world collections. It is shown that SARCASTIC can be used to better understand such phenomenology and even, confirm the source of multipath and vibration signatures when present in SAR imagery. Finally a new approach to the detection of vibration signatures is presented together with evidence that vibration signatures can have asymmetrical signatures in some cases.

1.1 Novel Contributions

The following are the novel contributions of this research:

- The thesis develops the theory of electromagnetic scattering in terms of electrical resistivity and magnetic permeability from first principles.
- A novel approach to the generation of a hierarchical data structure called a *kd-tree*, used for fast ray-tracing is presented and is shown to outperform previous ray-traversal techniques.
- A SAR ray tracing simulator that allows imagery analysts within the intelligence community to gain an understanding of their SAR imagery has been developed.
- An approach to gain an understanding of the structure of features inside a building by inverting the ray-tracing process called *ray-unravelling* is demonstrated.
- A technique for the detection of vibration signatures is developed and then validated using real SAR imagery from an experimental airborne SAR called *Bright Spark*.
- The work presents a way to visualise vibrating targets using a *multilook movie*.
- The existence of a unique SAR signature called a *one-sided paired echo* is theorised and then validated using an experimental indoor SAR system called *GB-SAR*.

1.2 Thesis Structure

The thesis starts with an overview of the evolution of synthetic aperture radar and describes the key phase terms present in the formation of a target within a SAR scene before proceeding to discuss the issues associated with SAR simulation and the development of a fast, but precise, SAR simulator.

The formulation of the SARCASTIC simulator requires detailed understanding of three separate scientific disciplines: an understanding of the electromagnetic interactions between a radar pulse and an object's surface; an understanding of the underlying primitives in computer aided design; and an understanding of the

interaction between these two areas in the form of electromagnetic ray-tracing. For this reason Chapters 2 through Chapter 4 individually address the required background material before the design approach to SARCASTIC is discussed.

During the formulation of the SARCASTIC algorithm it is important to have a thorough understanding of electromagnetic scattering, particularly from materials that are not considered to be a perfect electrical conductor (PEC). A common approach to SAR simulation and modeling is to assume that all radar interactions are with such PEC materials. This results in artificially bright returns in simulated SAR imagery and does not take into account penetration through a material. Chapter 2 therefore starts from first principles and derives the electromagnetic scattering equations required to simulate a wide range of materials with different electrical and magnetic properties.

The next important consideration for SAR simulation is the formulation of the CAD model against which a SAR image will be simulated. To be useful to an analyst they must be able to rapidly refine their CAD model and rerun the simulator. For this reason a simple triangular facet approach to CAD construction was taken that utilises an open standard and freely available CAD construction tools. Despite the availability of tools there are specific nuances required for SAR ray-tracing that are not useful to the wider CAD industry. Chapter 3 will cover these together with algorithms for the conversion from a CAD model into a *triangle mesh*.

The formulation of electromagnetic interaction and a well characterised triangle mesh are only the inputs to a simulator. A significant challenge arises in the projection of rays from the sensor through the mesh to determine the scattered electromagnetism or electromagnetic (EM) fields. This subject is not unique to SAR simulation and so Chapter 4 discusses how the computer games industry approaches this problem and specifically how it can be optimised using hierarchical data structures and the general purpose graphical processing unit (GPGPU). The chapter presents a history of such data structures and specifically how the implementation of a *kd-tree* can be used. It concludes by showing that SARCASTIC is able to ray-trace through a CAD model at least as quick as a high end (2017) computer

game without losing integrity with regard to ray-facet interaction.

Having the component parts to a simulator in place, Chapter 5 describes how they are engineered into the SARCASTIC program. There is an additional cost saving that can be realised through implementation if the SAR sensor is operating in Spotlight mode - that of pulse under-sampling which is also discussed.

Chapter 6 presents results using SARCASTIC to explore multi-path phenomena. In this chapter, real world data collected using the DLR F-SAR system is compared to the simulator. It is shown that the presence of features within a building can be determined through a *ray-unravelling* technique even though they cannot be directly observed from outside.

Finally Chapter 7 explores the rapid ray-tracing capabilities of SARCASTIC. Being able to rapidly rebuild a kd-tree data structure means that the CAD model can be rebuilt during simulation. This lends itself to the simulation of moving targets within the scene. Results are provided that show that SARCASTIC can simulate a vibrating generator. The signatures from such a generator are described in terms of their theoretical construction. It is shown that the operational status of such an object can be inferred in a real SAR image using the Thales/Dstl *Bright Spark* SAR sensor. The chapter concludes with a presentation of a useful way to identify vibration signatures in SAR imagery.

Conclusions are presented at the end summarising the areas discussed and making recommendations for further work and improvements.

1.3 The Evolution of SAR Image Formation

1.3.1 Optical Processing Origins

The concept of Synthetic Aperture Radar was first developed by Wiley in 1951 [1] working for the Goodyear Aircraft Corporation, Arizona. The aim of the work was to develop a radar technique that would provide a finer azimuth resolution than the azimuth extent of the radar beam. The work was designed for use with the US ATLAS ICBM programme [2]. In 1952, Sherwin et al. produced similar experimental results independently [3] although not published in the open literature

until 9 years later. In both cases the concept was similar. Radar pulses were transmitted from a moving aircraft in the direction perpendicular to the direction of flight and then the received radar returns were *range-binned* as with traditional radar systems (Figure 1.2). The innovation applied by Wiley and Sherwin was that the radar pulse returns could also be filtered using electronic filtering circuits in terms of the Doppler shift applied to targets within the beam caused by the motion of the platform. The initial concept was to observe only the zero-Doppler component of the radar beam which would yield an azimuth resolution equivalent to the Doppler width of the electronically filtered circuit. Wiley called this technique Doppler beam sharpening (DBS) and quickly realised that the fundamental limitation, that of the amplitude of the captured radar power, could be improved by coherently combining additional Doppler filter bands to the zero Doppler band. Wiley applied for a patent for this technique in 1954 and the technique was called 'unfocused' or 'semi-focused' radar.

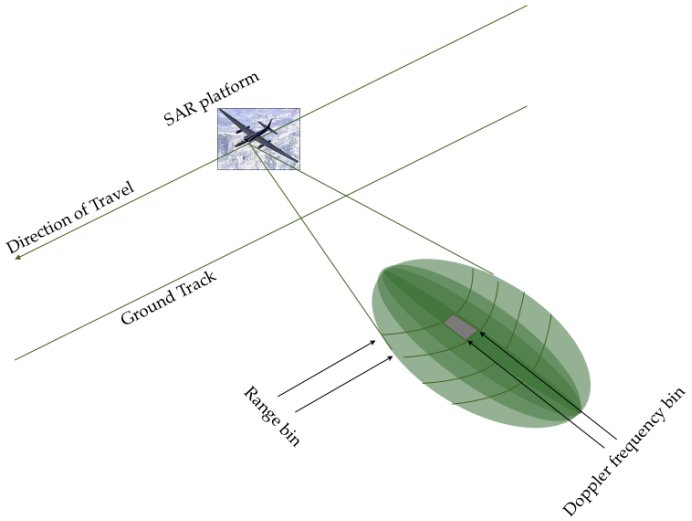


Figure 1.2: Operation of a SAR Sensor. Early SAR systems obtained azimuth resolution by filtering the received signal in terms of the Doppler shift caused by the motion of the SAR platform.

As a result of the success of this work, a project was funded by the US Department of Defense called the 'Side-Looking Radar Project'. The programme was run by the Willow Run Research Centre (WRRC) at the University of Michigan. Early

analysis by Dr. Louis J. Cutrona, Weston E. Vivian and Emmett N. Leith of the WRRC Radar Laboratory [4] theorised that it would be possible to produce a synthetic azimuth resolution equivalent to half of the radar antenna length, provided that the radar returns could be combined coherently using some yet-to-be-determined technique.

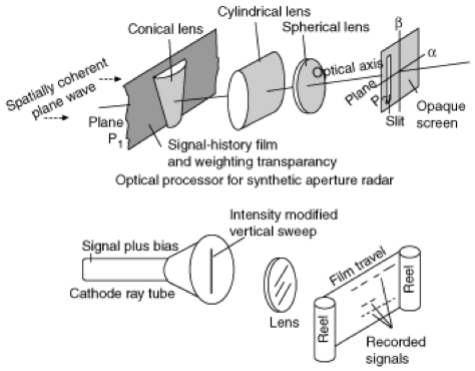


Figure 1.3: SAR Optical Correlator

At the time the various electronic Doppler filter bins were displayed across the width of a cathode ray tube and captured onto 35 mm film. The Radar Laboratory's principal innovation was to recognise that each scatterer within the scene had a similar history of Doppler frequencies, for a given range. By shining light rays through the 35 mm film and at the same time through a piece of 35 mm film with a pre-recorded signal history on it, the two beams could then be 'correlated together' to produce a coarsely focused image [5]. Later developments by Leith and his team [6] would recognise that the quality of the images, dependent upon the focal length of the zone plates used to correlate the radar returns with a known reference, depended on the accuracy of the wavelength of light used through the optical correlator. In order to obtain a stable monochromatic, coherent light source Leith initially used a mercury-vapour lamp and colour filter [7] and eventually a helium-neon laser light source [8]. It was not until August 1957 that a complete SAR system with optical correlator (Figure 1.3) was able to produce a fully focused SAR image.

1.3.2 Beyond Unfocused SAR

One of the necessities that forced the development of SAR was the requirement for finer resolution. It was recognised that coherently combining large numbers of pulses across the synthetic aperture caused additional defocussing of the SAR imagery due to the fact that individual radar reflecting objects now moved through

range samples. This motion through range cells (MTRC) required an additional range migration correction step to be performed. Initially this was performed through the incorporation of cylindrical and conical lenses [9, 10] within the processing chain.

During the 1960s and 1970s SAR processing improvements principally revolved around more sophisticated designs of optical radar correlators. The most basic image formation processors considered only a strip-map mode of image formation where the illuminated region of the imaged scene moves parallel to the sensor platform and at the same velocity [11, 12]. These systems usually had coarser range and azimuth resolution and so the digital radar pulses could easily have their motion through range cells and any required phase corrections applied (for example due to platform motion) [13].

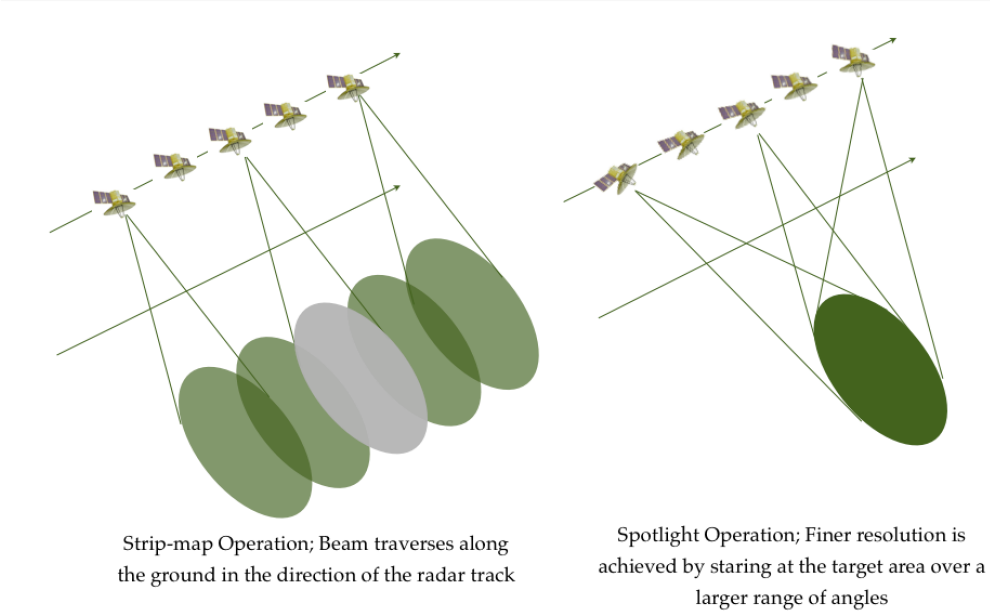


Figure 1.4: Comparison of Strip-Map and Spotlight Modes of SAR Operation

In a 'spotlight mode' of operation (Figure 1.4) the radar beam stares constantly at a patch on the ground whilst the platform transits past the target area (called *beam steering*). This allows a considerably larger range of Doppler frequencies to be sampled by the radar than those determined by the width of the illuminated radar beam. In doing so considerably more pulses could be coherently combined and so a finer azimuth resolution could be achieved.

A breakthrough was made by Walker [14, 15], working with Brown in studying the formation of imagery from a static radar looking at rotating objects. The original requirement was to use radar to image objects in space as they flew over the radar [16]. It was recognised by Walker that this experimental setup could also simulate a static target being imaged by a radar flying around the target area and was, in essence, the first spotlight-mode SAR. Traditionally the subsequent pulse returns were stored side by side in a rectangular format. Walker recognised that the image formation problem could be greatly simplified by storing the pulses in a polar format along radial lines from the instantaneous sensor position to the target area. This technique was patented by Walker and Carrara in 1980 [17].

1.3.3 Digital Image Formation

The mathematical requirements for image formation were quite considerable if performed on a computer system. Typically around 1000 floating point operations were required for each pixel produced and so a 1000x1000 pixel image would require approximately 1 billion calculations [18]. In contrast an optical correlator could achieve this image formation process almost instantaneously. It was not until the late 1970s that computers became fast enough that SAR images could be formed in a realistic timescale. Whilst still being slower than their optical equivalents, the engineering advantages of a software only solution were not missed on the military [19].

As a result of the advances in computer speed, the 1980s and 1990s made significant improvements in the quality of SAR image formation in addition to the amount of area that can usefully be imaged. Polar formatting and spotlight-mode imaging became the technique of choice for the military providing high azimuth and range resolution. The polar formatting technique however did suffer from limitations. In addition to the desire for finer resolutions, there was also a desire to increase the size of the area on the ground being imaged. In order to correctly focus a target within the scene the correct Doppler phase history of the target has to be accounted for. Any mismatch between the actual phase history of a target and that of the reference function used for correlating the target, results in defocussing. Whilst the polar

format algorithm (PFA) was able to focus a target at scene centre with no residual phase errors it assumed that all scatterers within the scene had a 'polar' phase history. Scatterers away from the scene centre however were shown to have non-polar phase histories (analogous to the comparison of planar vs non-planar radar wavefronts [20]). Roca, Prati and Cafforio developed a new approach in 1988, adopted from the field of seismic imaging that accounted for the curved wavefronts of the incident radar pulses. [21,22]. The approach, known as the range migration algorithm (RMA) can be applied to stripmap and spotlight mode processing and differs in principal from the polar format algorithm in that it applies a phase correction to each radar pulse such that they appear as if collected from a constant straight line trajectory. (conversely the polar format algorithm naturally corrects each pulse to a point or the scene centre). In addition the RMA incorporates a 'Stolt Interpolation' process [23] that applies a correction to the wavefront curvature via a 2 dimensional Doppler frequency / radar frequency dependent phase correction. The RMA technique however has two principal limitations. Firstly the requirement to motion compensate to a straight line means that the radar collection has to be up-sampled in order to adequately support the, now, much higher Doppler frequencies subtended by the synthetic aperture. In the polar format approach, motion compensation to a scene centre point means that only the relative, instantaneous difference between the Doppler frequencies imaged and that of the scene centre have to be sampled. This means that the RMA requires significantly larger memory requirements than the polar format algorithm.

The second limitation of RMA is that of *residual video phase*. In order to circumvent the requirement for very high digitisation rates for high resolution radar systems, a technique called *deramp-on-receive* was developed whereby the inbound radar pulses are mixed with a reference pulse with the conjugate of the expected signature from a target at scene centre. This is performed prior to digitisation in the radar receiver and reduces the required digitisation bandwidth to that of the difference between the received radar returns from the target and those from the scene centre (in the same way that the polar format algorithm reduces the along track sampling requirements by motion compensating the radar returns to scene centre).

A knock-on effect of this process is that although targets at different ranges across the swath are now discriminated by the differences in their deramped frequencies, it does not correct for the issue that radar pulses from targets nearer the sensor arrive before those that are further away. This effect is called residual video phase (RVP) [24]. The magnitude of this effect however is proportional to the difference between the range from the sensor to a reflecting target and that of the range to the scene stabilisation point. For PFA this difference is usually relatively small - the target usually being quite near to the scene centre. For RMA however the difference in range between a target and the motion compensation point, usually the 'stand-off range' for RMA, can be larger - particularly at higher squint angles. As a result RMA usually has to correct for RVP when employed with a radar system that applies deramp-on-receive to its inbound pulse returns.

During the 1980s two computationally efficient variations of the RMA were developed. The first, called chirp-scaling reduced the requirements for range interpolation through the application of phase corrections to the chirp waveform. This significantly reduces the computational load of the algorithm but relies on the received radar pulse being stored as a chirp waveform rather than being a deramped version of the transmitted chirp. Low resolution commercial SAR systems usually digitise the radar pulse at the radar front end as the radar's Analogue to Digital converters are often fast enough to digitise the relatively low bandwidth of the radar chirp waveform. This situation rarely occurs however for high resolution military SAR systems that trade digitiser bandwidth for radar swath width and so deramp the radar pulse at the receiver. In these situations the chirp has to be reapplied to radar range data which adds an additional computational load to the algorithm. Another limitation of the chirp scaling algorithm (CSA) is that the phase corrections do not adequately compensate for higher order range-rate terms and so, similar to the polar format algorithm, there is a maximum scene size that can be processed using CSA.

The second technique, tomographic back projection (TBP) was based upon computer aided tomography (CAT) signal processing to reconstruct the required wavefront curvature within the synthetic aperture data set. The technique recon-

structs a representation of the entire SAR scene for each pulse received and coherently combines additional pulses (with correct phase adjustments) to the original scene. This technique has two distinct advantages - it allows a SAR scene to be built up in a piecemeal way such that downstream algorithms such as automated target detection / recognition can start working on the early lower resolution data immediately and test / validate their results as more pulses are added to the scene. The second advantage of the TBP algorithm is that it is very easy to 'parallelise' on massively parallel architectures such as GPU (Graphics Processing Units) systems [24–26]. Unfortunately comparison of the computing resources required for the TBP algorithm with RMA are not favorable with RMA being slightly more efficient in its implementation.

There does exist however, a SAR processing algorithm that completely corrects for all targets within a scene with no residual distortions or phase errors. This is known as the time domain algorithm (TDA) [27]. This technique processes each pixel within the final SAR image by identifying the individual radar samples within the collected synthetic aperture and correctly integrating them together. As a result it is incredibly resilient to spatially variant image defocussing effects (such as variations in terrain height, atmospheric perturbations or even unaccounted for platform motions). The technique is incredibly computationally expensive however, requiring each pixel to be accounted for independently. Despite this, developments in massively parallel computer architectures have enabled time domain processors to start being employed [24, 25, 28].

Over the last 60 years SAR image formation has evolved from optical correlators to ultra fast digital signal processing. In such time there have been many significant breakthroughs in the understanding and optimisation of approaches that adequately approximate the physical processes involved. With the advent of massively parallel computer architectures such as those demonstrated on NVIDIA graphical processing unit (GPU)s, the SAR processor has now evolved such that approximations and shortcuts are no longer required to produce timely SAR imagery. The time domain processor has now come of age and with it the ability to correctly process every pixel within an image with the introduction of no artifacts from the image processor.

Images formed under such conditions are therefore perfect for detailed and precise analysis of artifacts caused by atypical scattering mechanisms within the scene being imaged. This is therefore the subject of the next section.

1.4 Atypical Scattering in SAR Images

All existing SAR image formation algorithms make a series of assumptions about the nature of the scene being imaged. These assumptions are:

- A target within the scene is stationary during the synthetic aperture duration
  - In terms of its micro-motion - vibration
  - In terms of its macro-motion - target displacement.
- A target's reflection properties do not vary with instantaneous direction of illumination
- All received signals within a scene come from a single target bounce (no multi-path)

In addition it is almost always assumed that:

- there is no motion of the platform in the time between a pulse being transmitted and it being received (the start-stop approximation)
- A target's frequency response is constant across the frequency modulated pulse.
- The propagation medium through which the pulse passes is homogeneous.

These assumptions are incorrect to varying degrees for a range of target imaging scenarios. When a target violates one of the assumptions above, it is incorrectly represented by the image formation algorithm. This causes the SAR image to contain unwanted artifacts that reduce the perceived quality of the SAR imagery. It is postulated however that by studying these atypical scattering radar-target interactions, additional information about the nature of the target being imaged can be

gleaned. The problem therefore is one of identifying the nature of the violation from the incorrectly focused SAR response of the target. The following section will discuss the research made to date on each of these areas.

1.4.1 Target Motion

The move to finer resolutions has significantly increased the duration of the synthetic aperture being collected. It is little wonder therefore that the most obvious form of atypical target scattering is caused by motion of a target whilst it is being imaged. In this document two types of target motion will be considered; macro-motion - that of displacement of a target during the synthetic aperture collection period and micro-motion, characterised by the vibration of a target but not its displacement. The former topic has been considered in detail whilst the latter is a relatively new area of SAR research.

One of the earliest publications on the effects of target displacement in SAR imagery was published by Graf and Guthart in 1969's IEEE Transactions on Antennas and Propagation [29]. This early text only focused on the first order effects of the along track motion of the target relative to the sensor, namely the velocity and did not consider higher terms of target motion. It suggested that the dominant effect of motion was uncompensated radial motion that leads to a displacement of the target whereas motion in the along-track direction is a secondary effect causing target defocussing. It was suggested by the authors that an iterative approach using estimates of velocity could accurately determine the target motion and therefore provide corrected and focused SAR imagery. This approach was extended by R. K. Raney [30] who constrained the problem to slow moving targets with constant velocity. A breakthrough occurred in 1985 when Ouchi [31] adopted the principle of multi-looking SAR image formation to better characterise a moving target as the sensor transits past. This was extended further, by many authors [32–38] and demonstrated primarily on SAR imagery of ships due to the large difference in signal strength between the moving target and background ocean. It should be noted that these techniques all focus on single phase centre image formation. The detection of moving targets using multiple-phase centers and the application of Space-Time

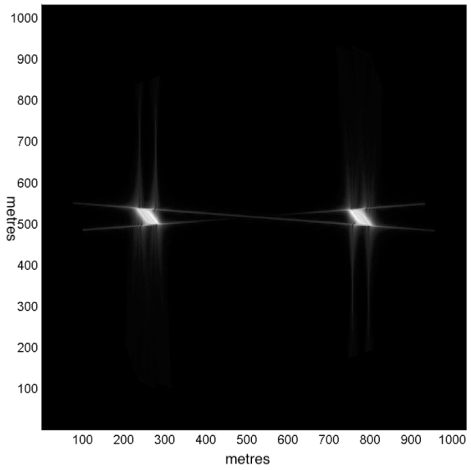


Figure 1.5: SAR Amplitude Modulation. Simulated SAR image of a point target that has amplitude modulation whilst being imaged. The targets signature is represented by two displaced *fuzzballs* located either side (azimuth direction) of the true target location

adaptive Processing (STAP) algorithms such as the Displaced Phased Centre Array (DPCA) are well demonstrated e.g. [39] are not considered in this document.

The accurate characterisation of moving targets within single phase centre SAR imagery is still being researched but recent work in this area [40–42] has now focused on determining why this has been such a challenge and has concentrated on identifying the limited scenarios for which a solution can be found.

One of the earliest documented approaches to characterising micro-target motion effects within SAR imagery is from Subotic et al. [43] and discusses how a phase modulating target signature might be detected within SAR imagery from the appearance of 'fuzzballs' or defocussing *paired echoes* that appear either side of the moving target in the azimuth direction (Figure 1.5). A later work [44] looked at the overlap between the visible characteristics of modulating signatures and compared them to static rotating targets such as rotating antennas. This work was applied to real data from the *MEMPHIS* millimeter wave system but concentrated more on the rotation effects than the modulation effects.

Recent work in this area has focused on the theoretical understanding of such effects [45–47] but due to the requirements for radar power and fine resolution, there have been no publications on the techniques applied to real data prior to this thesis.

1.4.2 Changes in Target Radar Cross Section

The SAR IFP assumes that all targets within the scene have a constant radar cross-section throughout the synthetic aperture collection. It is clear that this assumption becomes increasingly invalid as the range of imaging angles that the SAR images a target from increases. A special case of this effect is closely aligned with the above consideration of micro-motion - that of amplitude modulation. Originally this was apparent in SAR imagery as an unwanted image formation effect as SAR image formation processors failed to adequately correct for the transition from one range resolution cell to the next [12]. Later it became apparent that a constantly modulating target radar cross section (RCS) could also produce the effect and that if correctly characterised may provide useful information regarding the target being imaged [47,48]. Regarding the more general case of varying radar cross section, the majority of the work in this area has focused on the determination of sea-clutter statistics [49–52]. Very little work has been performed on the general characterisation of RCS varying effects on land-target structures.

1.4.3 Radar Multipath

The term *multipath* refers to the indirect radar pulse returns from a target. They are characterised by having 1 or more reflections in addition to the normal radar-target interaction. These reflections introduce additional delays to the expected return signal from a given target. These delays are then characterised as a ghost signature of the original target in the downrange direction. Initial research in the area of characterising multipath signatures was focused primarily on the removal of such signatures [49, 53] as they caused unwanted artifacts within the SAR scene. Later work branched into two areas, that of modeling the SAR multipath effects [54,55] and that of developing algorithmic techniques to characterise the real world multipath effects that occur within actual, usually high resolution, SAR imagery [56–59]. A limitation of both of these approaches is that they consider the radar-target interaction from the perspective of a formed image rather than as a series of, possibly erroneous, radar target interactions as suggested in [49].

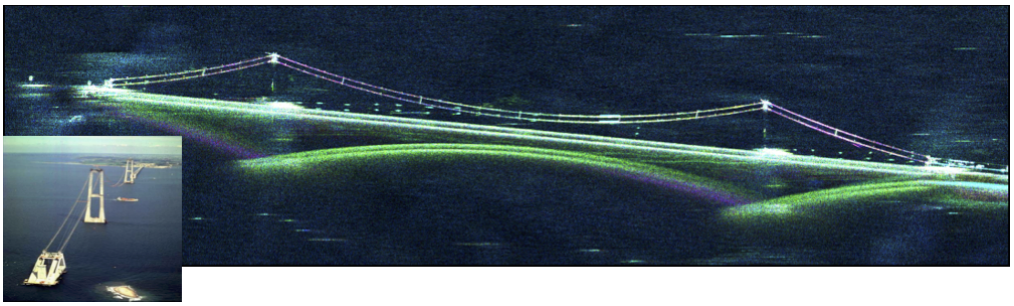


Figure 1.6: SAR Multipath in Action. EMISAR - Danish C-Band Polarimetric SAR system showing multipath effect from the Storebaelt Bridge taken 19th June 1996. Note the roadway on the bridge has not yet been built (inset) but appears complete on the SAR image from the multipath interaction between the hanging support lines and the water's surface. Image courtesy Technical University of Denmark

1.4.4 Start-Stop Approximation

In conventional SAR analysis, a model is developed that assumes that the sensor is stationary for the transmit period and receive period of each pulse. After receiving a pulse it is assumed that the sensor moves to the next transmit-receive position. For short pulse durations and relatively slower platform velocities, this approximation was shown to be adequate [60]. When the velocity becomes larger, for example a spaceborne platform, or when the pulse length becomes very large, for example a high mean-power system them two affects were predicted [61]. The first effect is caused by the motion of a platform during the intervening pulse round trip and was suggested by Tsynkov [62] to be insignificant in all cases of SAR image formation. The second effect is more pronounced and is caused by a Doppler shift being applied across the receive radar waveform caused by the platform motion. It was predicted in [62] that additional non-linear platform motion effects may have an additional impact on the quality of the SAR imagery.

1.4.5 Varying Frequency Response

The SAR image formation processor assumes that each target within an imaged scene has a constant radar cross section over the full bandwidth of incident radio frequency (RF) frequencies. This was an accurate assumption for early radar systems with only a few kHz bandwidth and a relatively high centre frequency, but modern systems attempt to maximise the bandwidth of the transmitted waveform in order to

achieve finer range resolution and experimental systems have pushed centre transmit frequencies lower in order to exploit the penetrative properties of these frequencies.

One of the first texts on the subject of frequency dispersion can be found in [63] which discusses how variations in RF response can cause range modulation effects within the compressed range signal. Additional texts [64–66] suggest techniques for identifying non-linear frequency responses and how to mitigate them within the hardware but do not discuss in detail how these affects are manifested within SAR imagery.

1.4.6 Inhomogeneous Propagation Medium

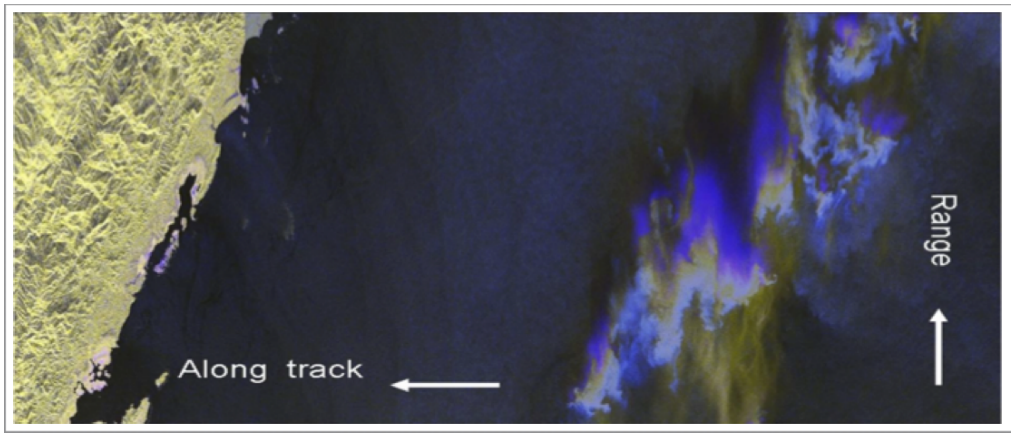


Figure 1.7: Propagation Effects on SAR Imagery. Taken from [67]. Colour composite HH/HH-VV/VV polarimetric image acquired by TerraSAR-X in strip-map mode over ocean surface close to the coast of Panama (Scene Center: Lat.: 5.5°, Long.: 62°). The image dimensions in range and azimuth are 30 and 60 km, respectively. The direction of illumination is according to the arrow given for the range direction in a descending orbit. The color coding is given as follows: (HH) red, (HH-VV) blue, and (VV) green. The blue-coloured zones correspond to regions, where the precipitation leads to pronounced differences in HH and VV signal intensity.

The assumption that the propagation medium for a SAR system is homogeneous is probably one of the widest areas studied for radar systems with papers reaching back to the late 1960s, e.g. [68]. The atmosphere has two notably different regions that contribute different effects to a space based SAR system. The first is the ionosphere. This is the region of space between 60 and 1000 km above the Earth's surface and consists mainly of ionised particles. This ionisation has a range

of effects on the propagation of a radar pulse such as scintillation, Faraday rotation (FR), refraction, diffraction, absorption, noise emission, ohmic losses, phase delays, spectrum spreading, and defocussing [67] It was shown in [69–71] that the ionosphere has little effect on SAR systems above 3 GHz.

The troposphere is the region from the surface of the Earth to about 12 km in altitude. In this region, the principal consideration is that of attenuation and absorption of RF waveforms. It was shown in [72–74] that the effect of the troposphere below 1 GHz was negligible. Recent developments however, most notably with the launch of TerraSAR-X and presented in [67] has shown that at 10 GHz, the troposphere can have a distinct effect in the attenuation of the SAR signature and can lead to occluded areas within the scene, see Figure 1.7.

1.5 From Radar to Synthetic Aperture Radar

1.5.1 Introduction

This section will describe the fundamental processes of a radar system. A derivation of the matched filtering process will be provided in order to understand how modern radar systems have adapted their signal processing and hardware electronics to finer resolutions. The section will then develop the analysis of a synthetic aperture and consider the steps required for image formation. Finally the advancements made for spotlight mode SAR processing will be detailed and provide a characterisation of a radar signature collected by such a system.

1.5.2 Components of a Radar System

Figure 1.8 shows the key components of a radar system required for accurate signal generation. Once the system is tasked, position information drives the selection of the most appropriate waveform which is then provided to the transmitter. The positioning information is also used to select when the transmitter should transmit a signal. The signal to be transmitted is presented to the antenna via a circulator which isolates potentially high power outbound signals from lower power inbound reflected radar pulses. The inbound pulses are collected by the receiver which is slaved to the transmitter via a central timing generator. In this way the relative phase

between transmission of a signal and reception can be accurately measured. The inbound signal, whilst being coherent with the outbound signal is still centred at raw radio frequencies, usually many Gigahertz or even tens of Gigahertz. These frequencies are outside the range of modern analogue to digital converters and so a mixer mixes the inbound signal down to a sufficiently low intermediate frequency (IF) where its phase and amplitude is measured and quantized and provided to a high data rate digital recorder. The data on the recorder can now either be processed on-board the sensor platform or provided off-line to a processor (in the case of SAR an Image Formation Processor or IFP) where it is presented for exploitation and possible re-tasking.

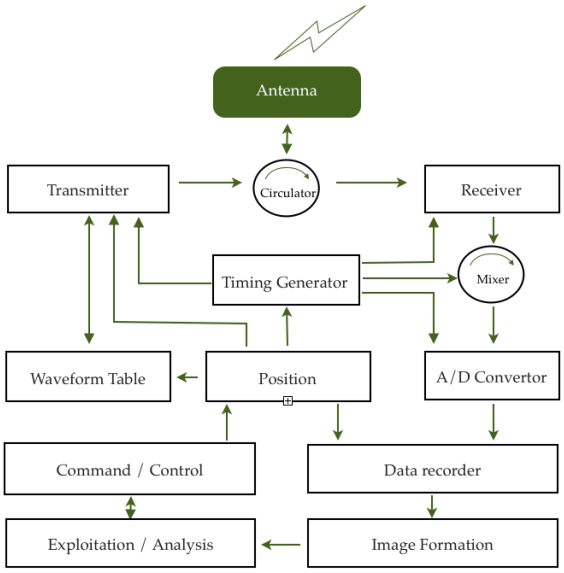


Figure 1.8: Block Diagram of Key Components to a Radar Sensor.

1.5.3 Single Pulse Physics Model

It can be seen from the above block diagram of a radar system that a critical component is the transmitted radar waveform and the waveform generator. As the energy from a transmitted pulse reduces as a function of the inverse square law, signals received back at the transmitter are reduced by $1/R^4$. The original approach to this problem was to increase the duration of the transmitted pulse. Long pulse durations however produce coarser resolution due to the requirement to resolve each pulse individually in space. The solution to this came with the

development of the linear frequency modulated (FM) waveform. This approach is widely used for modern day radar sensors and so will now be considered as a first step in the development of a fine resolution SAR model. Figure 1.9 presents a graphical depiction of a typical train of radar pulses. Each pulse contains a range of frequencies that varies linearly as a function of time. This type of modulation is known as a linear frequency-modulated pulse or more colloquially known as a *chirp* due to the way it sweeps from low to high (or high to low) frequencies in the same way as a bird's chirp sweeps over audio frequencies. Almost all modern

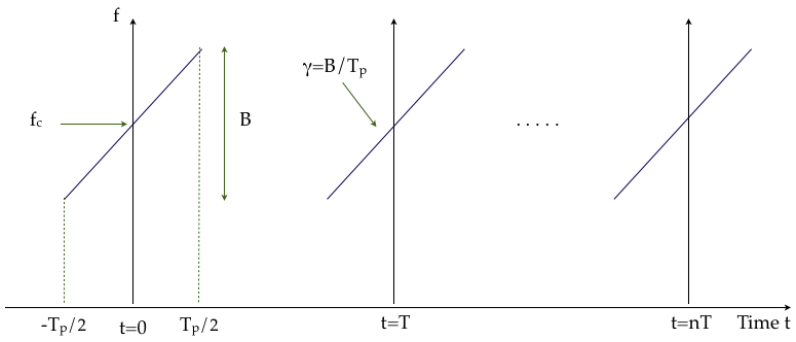


Figure 1.9: Diagram of a Pulse Train Transmitted From a Radar

radar systems transmit a linear frequency modulated chirp waveform centred around a frequency f_c and with a bandwidth of B and linear frequency modulation rate of γ given by:

$$\gamma = \frac{B}{T_p} \tag{1.1}$$

where T_p is the duration of the transmitted pulse. The frequency of the pulse at any given time t is therefore given by:

$$f(t) = \gamma t = \frac{B}{T_p} t \tag{1.2}$$

Upon collection back at the radar, the receiver is only able to measure the instantaneous frequency by measuring the instantaneous phase of the reflected signal (by comparing it to a known reference phase generated by the radar timing generator) and tracking the rate of change of this phase. In other words the measured phase

within the radar digitiser relates to the frequency modulated signal by:

$$f(t) = \frac{d\phi}{dt} \tag{1.3}$$

and so

$$\phi(t) = \int f(t)dt \tag{1.4}$$

Performing this integration on (1.2) provides the instantaneous form of the transmitted signal as measured by a radar receiver:

$$\phi(t) = \frac{B}{2T_p}t^2 \tag{1.5}$$

A problem now presents itself in the adequate digitisation of this signal. As the receiver only measures the voltage of the inbound signal, the analogue to digital (A/D) converter is required to have a sampling frequency of greater than twice the bandwidth B [75]. This is because only the real component of the complex EM signal is actually being measured. In order to determine the direction of rotation of the EM field phasor associated with the inbound signal it has to be measured more that twice per 2π rotation or greater than twice the pulse bandwidth. This causes a problem for the already bandwidth limited A/D converter whose principal limitation is the number of RF samples that can be measured per second. The most common solution to this is to employ digital quadrature demodulation between the mixer and introduce a second A/D converter. A representation of this processes can be seen in Figure 1.10

Figure 1.10 replaces the mixer and A/D Converter components of Figure 1.8 and allows the complex RF signal to be reconstructed without increasing the digitisation bandwidth requirements of the A/Ds within the system. In Figure 1.10, the inbound signal is demodulated (mixed) with a reference signal that matches the centre frequency f_c of the transmitted pulse. At the same time the reference signal is sent through a 90° phase shifter and again mixed with the inbound pulse return. This creates two signals, one of which lags in phase by $\pi/2$ radians from the original. The down-mixed signals are then low-pass filtered to remove sideband artifacts and

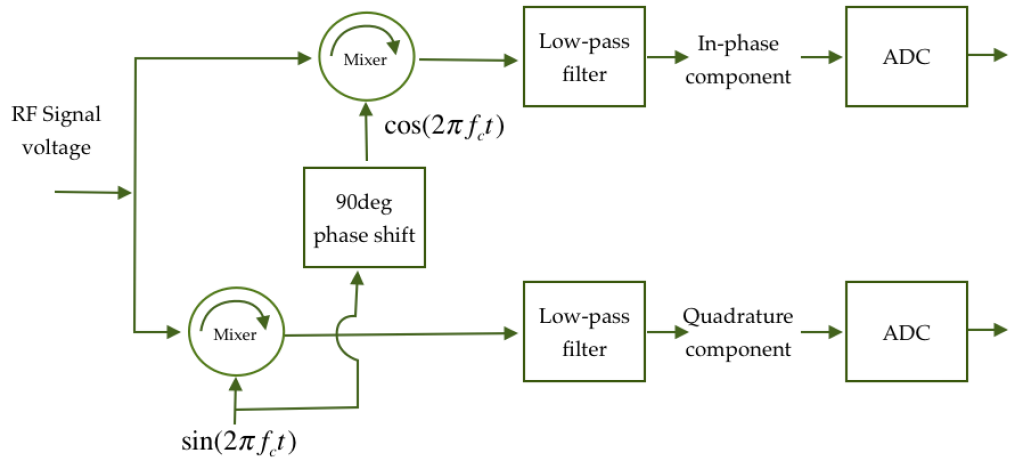


Figure 1.10: Complex Signal Reconstruction. (From [18] pp488). Complex signal reconstruction from real-valued voltage measurements of the inbound RF signal.

presented to the A/Ds at a reduced sampling rate of B samples per second. This is a considerable reduction in data rate from $2B$ or, the original requirement of $2(f_c + B/2)$ required for adequate sampling without demodulating to IF and digital quadrature demodulation.

At this point the radar records both the 'in-phase' component and the 'quadrature' component of the inbound signal. This allows the signal to be represented in complex form. A representation of the real or in-phase component can be seen in Figure 1.11

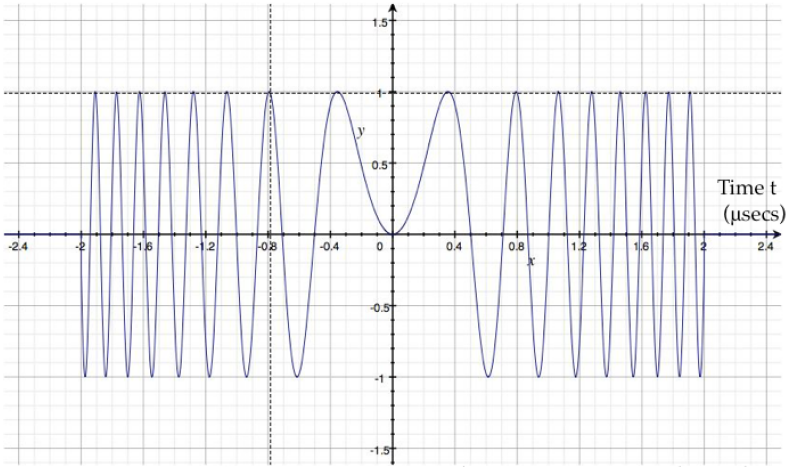


Figure 1.11: Representation of a linear, frequency modulated chirp waveform signature. In this case the pulse duration is $4\mu secs$ and the frequency modulation rate is 100 MHz/sec

The next component of the radar receive chain is the image formation processor. It is here that the inbound wide-band digital radar signal is traditionally compressed to a fine resolution pulse signal. In order to compress the pulse a copy of the outbound transmitted pulse is stored digitally and a matched filter generated such that the inbound radar return is convolved with the time-reversed conjugate of the reference. From (1.2), the frequency of the transmitted waveform as it exits the radar is given by:

$$f(n,t) = f_c + \gamma(t - nT) \tag{1.6}$$

where n specifies the transmitted pulse being considered with $n = 0$ being the first pulse in the pulse sequence. T is the inter-pulse period or the reciprocal of the pulse repetition frequency. By comparison of (1.2) to (1.5), the phase of this signal at any given instant is provided by integration of (1.6):

$$\phi(n,t) = 2\pi f_c t + \pi\gamma(t - nT)^2 \tag{1.7}$$

and so the transmitted waveform can now be represented in complex exponential form. Most radar systems find it convenient to define the envelope of the transmitted pulse by a rectangular function:

$$rect(u) = \begin{cases} 1, & |u| \leqslant \frac{1}{2} \\ 0, & |u| > \frac{1}{2} \end{cases} \tag{1.8}$$

Using this and defining $\hat{t} = t - nT$, the transmitted pulse at any given time t by the radar can now be defined as:

$$s(n,t) = A(t)\, rect\left(\frac{\hat{t}}{T_p}\right) e^{i(2\pi f_c t + \pi\gamma\hat{t}^2)} \tag{1.9}$$

where $A(t)$ represents the amplitude of the transmitted signal as a function of time and T_p represents the duration of a pulse. This signal does not take into account amplitude variations within the chirp waveform generator or frequency dispersion effects within the waveguides of the radar.

This signal then reflects from a target[1] within the scene and returns back to the radar system after a delay of t_d and can be presented as:

$$s(n,t) = \sigma_t rect\left(\frac{\hat{t}-t_d}{T_p}\right)e^{i(2\pi f_c t+\pi\gamma(\hat{t}-t_d)^2)} \tag{1.10}$$

where σ_t is the radar cross section of the reflecting target. The matched filter can now be constructed by performing the convolution of the received signal with the time-reversed, complex conjugate of the reference signal:

$$s_{rc}(n,t) = s_{ref}(n,-t)^* \otimes s(n,t) \tag{1.11}$$

The essential component here is the development of the reference signal. As the transmitted waveform does not vary much over time, and indeed is intended to be identical from pulse to pulse, radar systems usually pre-store the transmitted waveform which takes the form from (1.5) of:

$$s_{ref}(t) = e^{i\pi\frac{\gamma}{2}\hat{t}^2} \tag{1.12}$$

The time reversed conjugate of the reference is therefore :

$$s_{ref}(t)^* = e^{-i\pi\frac{\gamma}{2}(-\hat{t})^2} \tag{1.13}$$

and so the compressed pulse is defined by:

$$s_{rc}(t) = s_{ref}(t)^* \otimes s(n,t) \tag{1.14}$$

Expanding the convolution over just the duration of the pulse removes the pulse

[1]Here it is assumed that a single target reflects energy back to the radar. In actuality the reflected pulse is the coherent summation of pulse reflections from many scatterers over the illuminated scene and within the collected range window.

index variable n for simplicity:

$$s_{rc}(t) = \int_{-\frac{T_p}{2}}^{\frac{T_p}{2}} e^{-i\pi\frac{\gamma}{2}(-\hat{\tau})^2} e^{i\pi\frac{\gamma}{2}(\hat{\tau}-t)^2} d\hat{\tau} \tag{1.15}$$

$$s_{rc}(t) = \int_{-\frac{T_p}{2}}^{\frac{T_p}{2}} e^{-i\pi\frac{\gamma}{2}\hat{\tau}^2} e^{i\pi\frac{\gamma}{2}\hat{\tau}^2} e^{i\pi\frac{\gamma}{2}t^2} e^{-i2\pi\gamma t\hat{\tau}} d\hat{\tau} \tag{1.16}$$

$$s_{rc}(t) = e^{i\pi\frac{\gamma}{2}t^2} \int_{-\frac{T_p}{2}}^{\frac{T_p}{2}} e^0 e^{-i2\pi\gamma t\hat{\tau}} d\hat{\tau} \tag{1.17}$$

Expanding the integration provides:

$$s_{rc}(t) = -\frac{e^{i\pi\frac{\gamma}{2}t^2}}{i2\pi\gamma t} \left(e^{-i2\pi\gamma t\hat{\tau}}\right)\Big|_{-T_p/2}^{T_p/2} \tag{1.18}$$

which can be simplified thus:

$$s_{rc}(t) = -\frac{1}{i2\pi\gamma t} \left(e^{i\pi\frac{\gamma}{2}t^2} e^{-i\pi\gamma t T_p} - e^{i\pi\frac{\gamma}{2}t^2} e^{i\pi\gamma t T_p}\right) \tag{1.19}$$

$$s_{rc}(t) = -\frac{1}{i2\pi\gamma t} e^{i\pi\frac{\gamma}{2}t^2} \left(e^{-i\pi\gamma t T_p} - e^{i\pi\gamma t T_p}\right) \tag{1.20}$$

$$s_{rc}(t) = -\frac{1}{\pi\gamma t} e^{i\pi\frac{\gamma}{2}t^2} \sin(-\pi\gamma t T_p) \tag{1.21}$$

$$s_{rc}(t) = T_p e^{i\pi\frac{\gamma}{2}t^2} \frac{\sin(-\pi\gamma t T_p)}{-\pi\gamma t T_p} \tag{1.22}$$

$$= T_p e^{i\pi\frac{\gamma}{2}t^2} \operatorname{sinc}(\pi\gamma t T_p) \tag{1.23}$$

$$\tag{1.24}$$

which provides the impulse response function of the matched filter as:

$$s_{rc}(t) = T_p e^{i\pi\frac{\gamma}{2}t^2} \operatorname{sinc}(\pi B t) \tag{1.25}$$

where it can be seen that the form of the matched filter is a $sin(x)/x$ or *sinc* function whose null-to-null width is $1/B$ in terms of time or, more appropriately as:

$$\delta_r = \frac{c}{2B} \tag{1.26}$$

in terms of range resolution. The form of the matched filter output can be seen in Figure 1.12

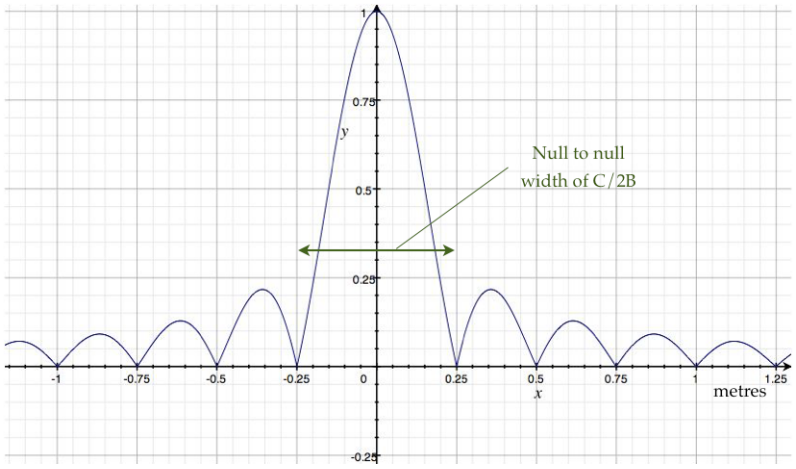


Figure 1.12: Matched filter impulse response for a linear FM chirp with a bandwidth of 600 MHz

1.5.4 Synthetic Aperture Analysis

Once the process for range compressing a single pulse has been developed, the next natural step is to consider a radar system flying at constant velocity past a target scene and transmitting a series of pulses to the side. Consider Figure 1.13. In this diagram the radar sensor constantly illuminates a swath parallel to the sensor track. The distance from ground track of the sensor to the start of the imaged scene is defined by the grazing angle g of the imaging geometry together with the sensor's altitude above the image plane. To develop the analysis of the aperture formation process consider a target T situated within the imaged scene at a broadside slant range of R_0 and at an along track location X_t. The sensor has a real aperture (antenna) dimension in the along track direction of d such that the antenna along track real

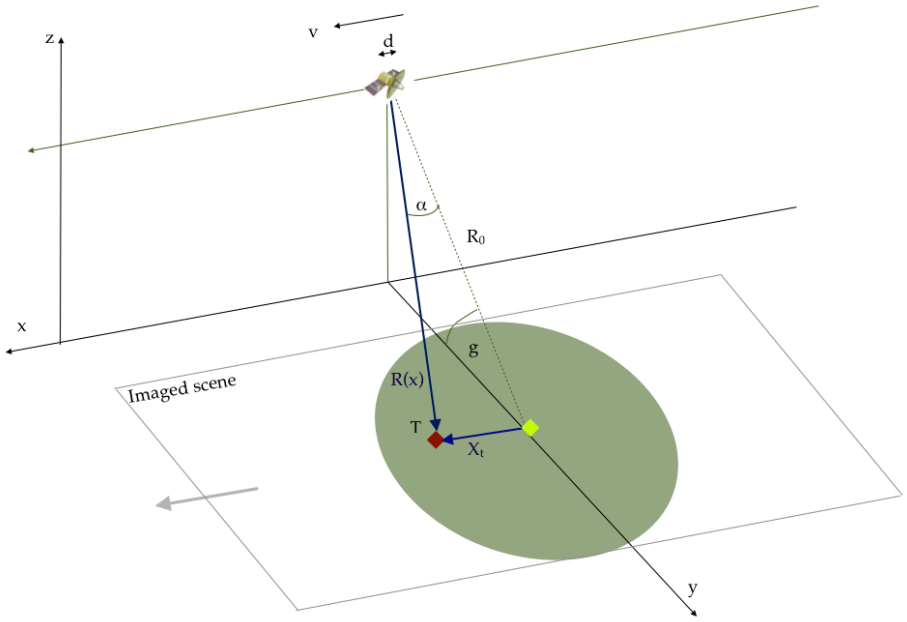


Figure 1.13: Geometry Description for Strip-Map SAR Operation

beam width is given by:

$$\theta_{beam} = \frac{\lambda_c}{d} \tag{1.27}$$

where λ_c is the centre wavelength of the transmitted chirp waveform corresponding to f_c. From conventional optics, the angular beamwidth would provide the angular resolution of the system and so the azimuth resolution of a real-aperture side looking radar would be provided by:

$$\delta_{ra} = R_0 \frac{\lambda_c}{d} \tag{1.28}$$

As the sensor transits past target T, the target enters the beam and remains illuminated by the beam until it exits at some time T_{sa} later. T is therefore illuminated over the synthetic aperture distance of L_{sa} which is the synthetic aperture length. During this transit time the radar pulse is observed by the target to undergo a Doppler frequency shift caused by the motion on the radar sensor. The cause of the Doppler shift is the platform velocity v which has a relative component of velocity towards the target of:

$$v_{rel} = v\cos\left(\frac{\pi}{2} - \alpha\right) = v\sin(\alpha) \approx v\frac{X_t}{R} \tag{1.29}$$

for small values of α. The Doppler frequency shift observed by the target is therefore:

$$f_d = \left(\frac{1 + {v_{rel}}/{c}}{1 - {v_{rel}}/{c}} \right)^{1/2} f_c \tag{1.30}$$

which by manipulation provides:

$$f_d = \left(\frac{1 + {v_{rel}}/{c}}{1 - {v_{rel}}/{c}} \right)^{1/2} \left(\frac{1 + {v_{rel}}/{c}}{1 + {v_{rel}}/{c}} \right)^{1/2} f_c \tag{1.31}$$

$$f_d = \left(\frac{\left(1 + {v_{rel}}/{c}\right)^2}{1 - {v_{rel}}^2/{c^2}} \right)^{1/2} \tag{1.32}$$

As $v << c$, v^2/c^2 tends towards 0 and so the observed Doppler shift at the target is:

$$f_d \approx \left(1 + \frac{v_{rel}}{c} \right) f_c \tag{1.33}$$

As this is the shift in Doppler frequency, the observed frequency by the target is :

$$f_{obs} = f_c - f_d = f_c - \left(1 + \frac{v_{rel}}{c} \right) fc = -\frac{v_{rel} f_c}{c} \tag{1.34}$$

Since the return echo will be shifted by the same amount

$$f_{radar} = -\frac{2v_{rel} f_c}{c} = -\frac{2vX_t}{R_0 c} \frac{c}{\lambda_c} = -\frac{2vX_t}{R_0 \lambda_c} \tag{1.35}$$

which provides a means of determining the location along track X_t of a reflecting target by measuring the Doppler frequency of radar pulses reflected from it.

The azimuth resolution of a synthetic aperture can now be calculated by determining the finest resolution of f_{radar}. By rearranging (1.35)

$$\delta_x = -\frac{\delta_{f_d} R_0 \lambda_c}{2v} \tag{1.36}$$

where δ_{f_d} is the resolution of the Doppler frequency shift that can be measured across the synthetic aperture and is approximately equal to the reciprocal of the time

that the target is being imaged (synthetic aperture duration T_{sa}) :

$$\delta_{f_d} \approx \frac{1}{T_{sa}} \tag{1.37}$$

As

$$T_{sa} = \frac{L_{sa}}{v} \tag{1.38}$$

and, for the case of strip-map imaging, the target is only illuminated whilst it is within the real aperture illumination beam pattern:

$$L_{sa(stripmap)} = \delta_{ra} = R_0 \frac{\lambda_c}{d} \tag{1.39}$$

$$\delta_{x(stripmap)} = \frac{R_0 \lambda_c}{L_{sa} 2} = \frac{R_0 \lambda_c}{2} \frac{d}{R_0 \lambda_c} = \frac{d}{2} \tag{1.40}$$

This provides a remarkably simple result for the case of strip-map imaging where the actual along track resolution of the system is inversely proportional to the antenna size of the sensor. Although this may seem counter intuitive at first, a smaller antenna has a wider beamwidth and so a target is illuminated across a much larger synthetic aperture. Practically speaking problems occur with very small antenna sizes due to the reduction is antenna gain. In addition, it was shown in [76] that for a synthetic aperture angle, α of π, the maximum possible, the resolution would approach $\lambda/2$. It was also noted in this paper that synthetic aperture angles greater than 60° are impractical and so a realistic limit to the along track resolution is actually $\lambda/4$.

1.5.5 Extension to Spotlight Mode SAR Image Formation

In addition to the practical considerations of antenna gain discussed above, there exists another limitation to producing finer azimuth resolutions from a strip-map sensor: that of range ambiguities and the pulse repetition frequency of the sensor. First consider the nature of range ambiguities (Figure 1.14). In this figure pulses are transmitted from the sensor with a fixed pulse repetition interval T such that the

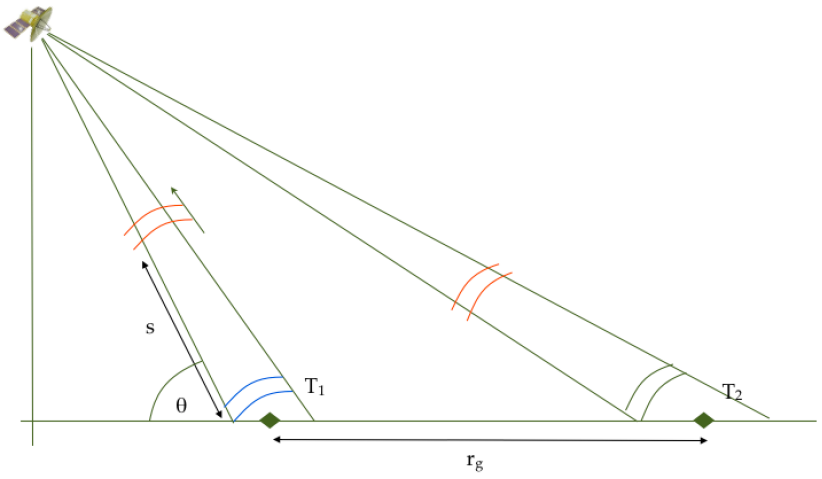


Figure 1.14: Range Ambiguities

distance in slant range between consecutive pulses s is:

$$s = cT = \frac{c}{prf} \tag{1.41}$$

with prf being the pulse repetition frequency. Pulses reflecting from target T_1 are separated in space by distance S. T_2 however is situated at a ground range distance from T_1 of :

$$r_g = \frac{s}{2}\cos(\theta) \tag{1.42}$$

such that direct returns from T_1 can continue to be reflected towards T_2 and return back to the location of T_1 within one inter-pulse period. When this occurs the reflected pulse from T_2 (red pulse in Figure 1.14) will arrive back at the receiver at the same time as the direct pulse return from T_1 (blue pulse in Figure 1.14). The radar receiver has no way of separating the two returns and tries to process them in an identical fashion. As the signature from T_2 has a different slant range, pulses from T_2 will not focus correctly when passed through the image formation processor resulting in ghost artifacts within the final scene.

From (1.41) it can been seen that increases to the pulse repetition frequency of the sensor will reduce the ambiguous distance between pulse returns and introduce more artifacts into the final scene. The prf however is a critical requirement

for adequately supporting the range of frequencies within the return pulse signal. From [75] and similar to the analysis for supporting the range chirp bandwidth, the sensor must sample the range of Doppler frequencies with a sampling frequency greater than the range of Doppler frequencies. Here the *sample* is actually the pulse and so this provides the constraint that:

$$prf > B_d \tag{1.43}$$

where B_d is the bandwidth of Doppler frequencies collected.

The range of Doppler frequencies of an SAR system is a function of the size of the illuminating beam in the along track direction and so to calculate the Doppler bandwidth consider the Doppler frequency of a target at the leading edge of the radar beam. From (1.35) the Doppler shift of such a point is:

$$f_d{}^{+} = \frac{2vf_c}{c}\cos\left(\frac{\pi}{2} - \alpha\right) = \frac{2vf_c}{c}\sin\alpha \tag{1.44}$$

and equally the Doppler shift from a target just departing the trailing edge of the radar beam is:

$$f_d{}^{-} = \frac{2vf_c}{c}\cos\left(\frac{\pi}{2} + \alpha\right) = -\frac{2vf_c}{c}\sin\alpha \tag{1.45}$$

as the Doppler bandwidth is the full range (positive and negative) of Doppler frequencies:

$$B_d = f_d{}^{+} - f_d{}^{-} = \frac{4vf_c}{c}\sin\alpha \tag{1.46}$$

In this case, as we are using the full extent of the radar beam:

$$\theta = \frac{\lambda_c}{d} = 2\alpha \tag{1.47}$$

$$\alpha = \frac{\theta}{2} = \frac{\lambda_c}{2d} \tag{1.48}$$

and so B_d, the Doppler bandwidth of the ground illuminated by the radar beam is:

$$B_d = \frac{4vf_c}{c}\sin\frac{\lambda_c}{2d} \tag{1.49}$$

and as λ_c is of the order of centimetres and d of the order of metres provides:

$$B_d \approx \frac{v}{\delta_x} \tag{1.50}$$

with δ_x being the azimuth resolution from (1.40). This equation is interesting as it shows that finer resolutions and faster platform velocities increase the range of Doppler frequencies illuminated by the sensor. From (1.43) however, this increase in bandwidth requires the *PRF* of the sensor to be increased which as we have seen in (1.41) increases the number of range ambiguities present in the scene.

The solution to this problem was developed by Walker and Brown [14] [10] and is called '*spotlight mode*' imaging. The innovation involves mixing the received pulse return with those returns expected from a scatterer at scene centre. Consider Figure 1.15. In this figure three targets are arranged across the scene to be imaged. The sensor images the scene in a spotlight mode of operation by constantly pointing its beam towards the scene centre. The synthetic aperture duration is given by:

$$T_{sa} = \frac{R_0 \theta}{v} \tag{1.51}$$

Each target within the scene is observed to undergo an approximately linear rate of change of Doppler frequency (a chirp) as the radar transits past it. The form of these histories can be seen in the bottom left graph in Figure 1.15. Each target has the same chirp rate which is provided by:

$$\gamma_a = \frac{B_d}{T_{sa}} \tag{1.52}$$

with the bandwidth B_d of each target being provided by (1.46).

From (1.46), (1.48) and (1.51):

$$\gamma_a = \frac{4v^2 f_c}{cR_0\theta} \sin\left(\frac{\theta}{2}\right) \approx \frac{2v^2 f_c}{cR_0} \tag{1.53}$$

Upon receiving the radar pulses they are mixed with a reference chirp signature which is equivalent to the Doppler history of a target at scene centre. This has the

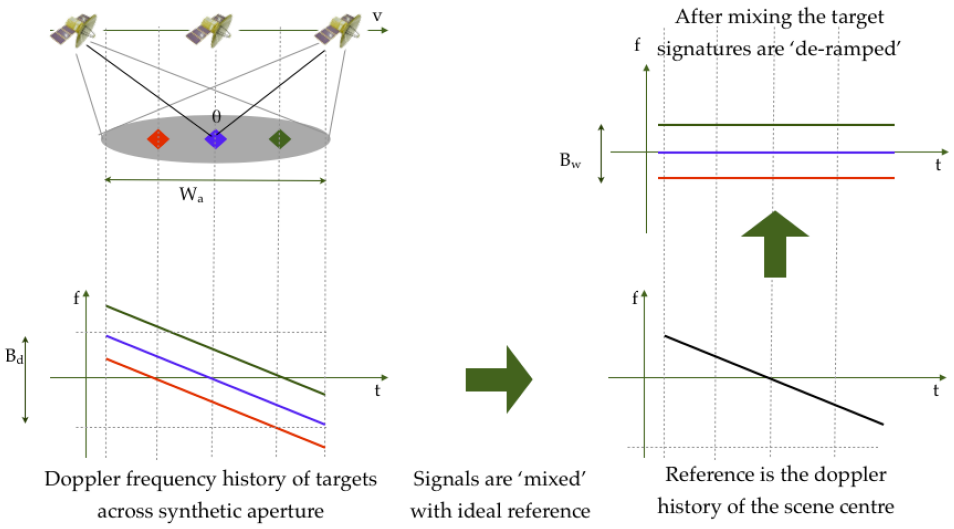


Figure 1.15: Azimuth Deramp Process Used in Spotlight Mode Imaging

effect of removing or '*de-ramping*' the chirp slope from the signature and effectively 'stablises' each target to a single frequency tone. Once stabilized, the imaged can be focused by taking the Fourier transform of the signal which forms a focused point for each frequency tone. This effect can be seen in the top-right hand graph of Figure 1.15. At this stage the bandwidth of the collected dataset is solely a function of the largest displacement from scene centre of a target. The bandwidth of the composite signal from all targets across a scene of width w_a is therefore the azimuth chirp rate multiplied by the time to traverse the scene:

$$B_a = t_w \gamma_a = \frac{w_a}{v} \frac{2v^2 f_c}{cR_0} \tag{1.54}$$

and so the pulse repetition frequency (sampling) requirements for a spotlight-mode SAR collection are

$$prf > B_a = \frac{2v f_c}{R_0 c} w_a \tag{1.55}$$

In effect the sampling requirements to support a very fine resolution has been traded for scene width w_a thereby allowing finer resolutions to be achieved than could be obtained through a conventional strip-map mode of operation.

1.5.6 Received Signal

Having constructed the theory behind SAR imaging and then extending it to fine resolution synthetic apertures, it is now possible to develop the form of the signal received by the radar. Before this occurs though there is an additional efficiency that can be made in terms of sampling frequency. The radar pulse waveform used is a linear FM signal which is very similar to the linear FM Doppler chirp generated through the aperture synthesis process. In the last section we showed how scene centre stabilisation performs a de-ramp of the Doppler chirp waveform into a constant tone. This technique can be applied to the radar pulse modulation. In this situation, the incoming radar pulse is mixed with the reference signature as it enters the radar receive chain and before it is digitised. In this way the wide-band pulse is reduced to a bandwidth determined by the range swath width in the same way as the wide-band Doppler signature of the beam was reduced to a bandwidth commensurate with the azimuth swath width. From (1.54) the bandwidth of the range de-ramped signal is provided by multiplying the chirp rate by the receive window duration:

$$B_r = \frac{\gamma 2 w_r}{c} \tag{1.56}$$

where w_r is the desired range swath width. By analogy to the azimuth case the sampling frequency required of the radar analogue-to-digital converter is now only a function of the range swath width and not the resolution (or transmitted pulse bandwidth). By employing such a 'de-ramp-on-receive' process and using the same reference signature for a stationary point at scene centre, the system is able to perform both range and azimuth stabilisation which significantly improves processing time and hardware requirements.

From (1.9) and recognising that the round trip delay time of a pulse from radar to a target at range R_t is:

$$t_d = \frac{2R_t}{c} \tag{1.57}$$

The signature received back at the radar is:

$$s(n,t) = A_t rect\left(\frac{\hat{t} - 2R_t/c}{T_p}\right) e^{i2\pi f_c\left(t - 2R_t/c\right)} e^{i\pi\gamma\left(\hat{t} - 2R_t/c\right)^2} \tag{1.58}$$

Before this is digitised, de-ramp-on-receive is performed by multiplying the inbound signal with the conjugate of the expected signal from the centre of the scene (range R_0). The form of the reference signal is therefore:

$$s_{ref}(n,t) = e\left[i2\pi f_c\left(t - \frac{2R_0}{c}\right) + i\pi\gamma\left(\hat{t} - \frac{2R_0}{c}\right)^2\right] \tag{1.59}$$

Multiplying these together provides the analytical form of the signal that is digitised:

$$s(n,\hat{t}) = A_t rect\left(\hat{t} - \frac{2R_t/c}{T_p}\right) e^{i\pi\left[2f_c\left(t - 2R_t/c\right) + \gamma\left(\hat{t} - 2R_t/c\right)^2 - 2f_c\left(t - 2R_0/c\right) - \gamma\left(\hat{t} - 2R_0/c\right)^2\right]} \tag{1.60}$$

$$s(n,\hat{t}) = A_t rect\left(\hat{t} - \frac{2R_t/c}{T_p}\right) e^{i\pi\left[-\frac{4f_c}{c}\left(R_t - R_0\right) + \gamma\frac{4\hat{t}}{c}\left(-R_t + R_0\right) + \gamma\frac{4}{c^2}\left({R_t}^2 - {R_0}^2\right)\right]} \tag{1.61}$$

which is more conveniently expressed as:

$$s(n,\hat{t}) = A_t rect\left(\hat{t} - \frac{2R_t/c}{T_p}\right) e\left[-i\frac{4\pi}{c}\left(f_c + \gamma\hat{t}\right)\left(R_t - R_0\right)\right] e\left[i\frac{4\pi\gamma}{c^2}\left({R_t}^2 - {R_0}^2\right)\right] \tag{1.62}$$

or

$$s(n,\hat{t}) = A_t rect\left(\hat{t} - \frac{2R_t/c}{T_p}\right) e^{-i\frac{4\pi\gamma}{c}\left(\frac{f_c}{\gamma} + \hat{t}\right)\left(R_t - R_0\right)} e^{i\frac{4\pi\gamma}{c^2}\left(\left(R_t - R_0\right)^2 + 2R_t R_0\right)} \tag{1.63}$$

the phase terms of which are given by:

$$\phi\left(s(n,\hat{t})\right) = e^{-i\frac{4\pi\gamma}{c}\left(\frac{f_c}{\gamma} + \hat{t} - \frac{2R_0}{c}\right)\left(R_t - R_0\right)} e^{i\frac{4\pi\gamma}{c^2}\left(R_t - R_0\right)^2} e^{-i\frac{8\pi\gamma}{c^2}{R_0}^2} \tag{1.64}$$

The reason for expressing the equation in this way is that it allows the received signature to be dissected in terms of components of image formation. These terms are summarised in Table 1.1. Of particular note is the 4th term in the table. This

Number	Term	Explanation
1	$\frac{4\pi\gamma}{c}\frac{f_c}{\gamma}\left(R_t - R_0\right)$	Doppler frequency term responsible for focusing a target in the along track direction
2	$\frac{4\pi\gamma}{c}\left(\hat{t} - \frac{2R_0}{c}\right)\left(R_t - R_0\right)$	Range frequency term. Responsible for focusing target in range direction.
3	$\frac{4\pi\gamma}{c^2}\left(R_t - R_0\right)^2$	Residual video phase caused by deramp-on-receive process.
4	$\frac{8\pi\gamma {R_0}^2}{c^2}$	Range dependent phase term that is invariant across the synthetic aperture and hence does not affect focusing in range or azimuth.

Table 1.1: Phase Terms Present in the Received SAR Signal

term is usually omitted in the standard texts (e.g. [18]) as it has no effect on the focusing of the final SAR image. It is important however when comparing phase effects between SAR images such as coherent change detection [77] as well as being an important term when comparing a SAR image with a simulation.

1.6 Summary

This chapter has provided an introduction to the thesis and described the evolution of SAR and the derivation of the key signal processing terms. The rest of this document describes how these principles can be expanded for SAR simulation.

Chapter 2

Background to Electromagnetism and Physical Optics

A key component of radar ray-tracing is the subject of electromagnetic field scattering from an illuminated target. An essential first step is therefore to present a thorough understanding of EM scattering in terms of the incident wave and the surface scattering properties. Ultimately this can be broken down into four simple steps:

1. The radar transmits an EM pulse
2. The pulse interacts with a scattering body
3. The EM pulse is scattered according to geometry and material properties of the body
4. Part of the scattered energy is received by a radar receiver

It therefore seems intuitive to break this down into two steps combining step 1 with 2 and step 3 with 4. This is the subject of this chapter and will be covered in three sections. For reasons that will become clear the first section will focus on steps 3 and 4 and the derivation of the scattered EM fields from an illuminated surface. It will be shown that the solution requires knowledge of surface currents induced upon the scattering medium.

Determining the surface currents from an incident EM field is considerably more complicated and will be covered in steps in the second section.

The second section will also discuss solutions to determining the induced surface currents and will derive a simple approach that can easily be applied to SAR simulation and modelling.

2.1 Derivation of Scattered EM Fields at the Receiver

There are many publications that describe the scattered electromagnetic field [78] [79] [80] [81] and many papers on the subject of radar simulation and modelling refer the reader to the relevant field equations referenced in one of the pre-eminent texts. In almost all cases the solutions are provided in general form and include descriptions for a wide range of EM field modes such as transverse magnetic (TM), transverse electric (TE) or transverse electromagnetic (TEM). It is also common to intersperse the derivation with examples of specific coordinate transformations such as cylindrical, rectangular and spherical coordinate systems. Another common feature of these texts are the numerous references to other works that are required to complete the physical understanding of the arguments being made.

Rather than simply refer the reader to these bodies of works, a complete derivation of the electromagnetic scattering equation is made here starting with Maxwell's Equations. The derivations made here differ from other work in this area in that they do not constrain themselves solely to electrical propertes of matter but also include those calculations that relate to magnetic conductivity. This is provided in order to offer a useful platform for further work related to modern meta-materials and allows the SAR simulator developed in the latter parts of this thesis to potentially simulate a wider range of materials than other simulations.

Whilst this derivation is certainly less general than the more notable works in this area, it is specific to the calculations of the far-field form of the scattered electromagnetic field and makes no loss of generality for polarisation or coordinate systems. It also attempts to fully capture the physics involved in the argument through various derivations and clarifications rather than referring the reader to other textbooks. In doing this it is hoped that a thorough understanding with regard to SAR ray-tracing will be developed.

2.1.1 The Approach

There are fundamentally two different approaches to developing the scattered field equations (Figure 2.1). The most obvious is described in [80] which is also adopted in [82] and develops Maxwell's equations in conjunction with the vector Green's theorem to arrive at a set of equations for the scattered field. This approach results in the Stratton-Chu equation, sometimes referred to as the Electric Field Integral Equation EFIE [83].

The second approach is the one adopted here and is described in [78], [79] and in [84]. Rather than performing the integration of the source currents to form the radiated fields, the approach first calculates the magnetic and electric vector potentials (A and F respectively) before relating them directly to the electric (E) and magnetic fields (H) through differentiation. In doing so the approach neatly circumvents difficulties caused in performing the integration of Maxwell's equations that can occur when searching for general solutions.

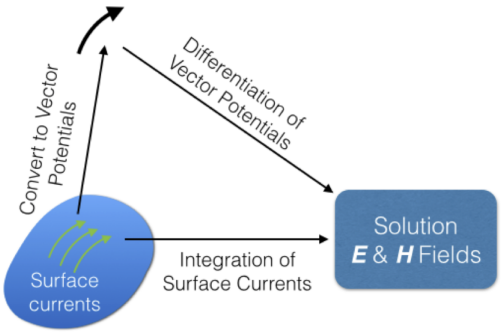


Figure 2.1: Different EM Approaches to Solving the EM Field Equation

Although the approach taken in [79] is detailed here, it is simplified to only consider transverse electromagnetic (TEM) waves in Cartesian coordinate space. The full derivation in [79] also covers transverse electric (TE) and transverse magnetic (TM) waves and also cylindrical and spherical coordinate systems but are not required for the analysis made here.

2.1.2 Defining the Fields in Terms of their Vector Potentials

The analysis starts, as always, with Maxwell's equations:

$$
\begin{aligned}
\text{(Gauss' Law)} \qquad & \nabla \cdot \vec{D} = \rho_e \\
\text{(Gauss' Law for magnetism)} \qquad & \nabla \cdot \vec{B} = 0 \\
\text{(Faraday's Law)} \qquad & \nabla \times \vec{E} = -\frac{\partial}{\partial t}\vec{B} \\
\text{(Ampère's Law)} \qquad & \nabla \times \vec{H} = \vec{J} + \frac{\partial}{\partial t}\vec{E}
\end{aligned}
\tag{2.1}
$$

Where the over-lined parameters ($\vec{}$) denote a vector. The meaning of each parameter and their units are defined in Table 2.1

Description	Formula	Units
Electric flux density	$\vec{D} = \varepsilon\vec{E}$	Cm^{-2}
Electric charge density	ρ_e	Cm^{-3}
Magnetic flux density	$\vec{B} = \mu\vec{H}$	Tesla or $N(mA)^{-1}$
Electric field	$\vec{E}$	Vm^{-1} or NC^{-1}
Magnetic field	$\vec{H}$	Am^{-1}
Electric current density	$\vec{J} = \sigma\vec{E}$	Am^{-2}
Permittivity (Dielectric constant)	$\varepsilon = \varepsilon_0\varepsilon_r$	Fm^{-1}
Permittivity of free space	ε_0	$8.854\times10^{-12}Fm^{-1}$
Relative Permittivity	ε_r	Always ≥ 1
Permeability	μ	$Hm^{-1} or NA^2$
Magnetic constant	μ_0	$4\pi\times10^{-7}Hm^{-1}$
Relative permeability	μ_r	Diamagnetic < 1
		Paramagnetic > 1
		Ferromagnetic $>> 1$
Electric charge density	$\sigma = \vec{J}/\vec{E}$	Sm^{-1}

Table 2.1: Definition of Parameters in Maxwell's Equations

As the following analysis is solely related to RF signals, it is useful to recast these into the time-harmonic form (and using the electrical engineer's definition of $j = \sqrt{-1}$ rather than the physicist's i).

$$\nabla \cdot \vec{D} = \rho_e \tag{2.2}$$

$$\nabla \cdot \vec{B} = \rho_m = 0 \tag{2.3}$$

$$\nabla \times \vec{E} = -\vec{M} - j\omega\mu\vec{H} \tag{2.4}$$

$$\nabla \times \vec{H} = \vec{J} + j\omega\varepsilon\vec{E} \tag{2.5}$$

Where $\omega = 2\pi f$ is the angular frequency of the signal being studied. In these equations we have introduced a 'magnetic charge density' ρ_m and 'magnetic current density' $\vec{M}$. This is common practice in a lot of derivations as, although no magnetic monopole has ever been discovered, it assists in the development of EM fields from surfaces that cannot be considered as perfect electrical conductors (PEC) because it allows magnetic current densities to be considered as the EM dual of the electric current density. This is mathematically useful for solving the equations for materials with more realistic permittivities and permeabilities. The first step is to develop the vector wave equation for EM propagation. To achieve this the wave equation in terms of the magnetic vector potential A and the electric vector potential F will first be developed. A and F are used extensively in antenna radiation problems [79] but it is worth recognizing that although the electric and magnetic field intensities (E and H) represent physically measurable properties, for most engineers the vector potentials are strictly mathematical tools. To understand this, consider the situation where there are no sources. In this case $\vec{J} = \vec{M} = q_e = q_m = 0$. In this situation the magnetic flux density is always solenoidal. i.e. $\nabla \cdot \vec{B} = 0$ in equation (2.3)

Recognising that the divergence of the curl of any vector is always exactly zero, in other words that for any given arbitrary vector, say $\vec{A}$:

$$\nabla \cdot \left(\nabla \times \vec{A} \right) = 0 \tag{2.6}$$

It is therefore possible to state that from equation (2.3)

$$\nabla \times \vec{A} = \vec{B}_A = \mu\vec{H}_A \tag{2.7}$$

and so

$$\vec{H}_A = \frac{1}{\mu}\left(\nabla \times \vec{A}\right) \tag{2.8}$$

This is the definition of the magnetic vector potential $\vec{A}$, namely that it is the field whose curl is defined as the magnetic flux. It is often used in textbooks as a purely mathematical construct to facilitate the solving of field integral equations. In actuality the $\vec{A}$ field is a real field and has been shown to exist in quantum mechanics through experimentation [85]. The subscript A denotes that the field is due to the magnetic vector potential. This can now be substituted into Faraday's law (2.4) (remembering that this is a source-free region with no currents flowing) and provides:

$$\nabla \times \vec{E}_A = -j\omega\mu\vec{H}_A \tag{2.9}$$

$$\nabla \times \vec{E}_A = -j\omega\left(\nabla \times \vec{A}\right) \tag{2.10}$$

$$\nabla \times \vec{E}_A + \nabla \times j\omega\vec{A} = 0 \tag{2.11}$$

$$\nabla \times \left(\vec{E}_A + j\omega\vec{A}\right) = 0 \tag{2.12}$$

and as the curl of the gradient must always be zero, i.e.

$$\nabla \times \left(-\nabla\phi_e\right) = 0 \tag{2.13}$$

it follows that

$$\vec{E}_A = -\nabla\phi_e - j\omega\vec{A} \tag{2.14}$$

which relates the electric field caused by the magnetic vector potential $\vec{A}$ to an arbitrary electric scalar potential ϕ_e that is a function of position. The vector identity

$$\nabla \times \nabla \times \vec{A} = \nabla\left(\nabla \cdot \vec{A}\right) - \nabla^2\vec{A} \tag{2.15}$$

can now be used with (2.7) to give

$$\vec{\nabla} \times \left(\mu\vec{H}_A\right) = \nabla\left(\nabla \cdot \vec{A}\right) - \nabla^2\vec{A} \tag{2.16}$$

which for a homogeneous medium (i.e. constant μ) can be reduced to

$$\vec{\mu} \nabla \times \vec{H}_A = \nabla \left(\nabla \cdot \vec{A} \right) - \nabla^2 \vec{A} \tag{2.17}$$

This can now be related to Ampère's Law (2.5) to give

$$\vec{\nabla} \left(\nabla \cdot \vec{A} \right) - \nabla^2 \vec{A} = \mu \vec{J} + j\mu\omega\varepsilon\vec{E} \tag{2.18}$$

Substituting the expression for the $\vec{E}$ field (2.14) into (2.18) gives

$$\nabla \left(\nabla \cdot \vec{A} \right) - \nabla^2 \vec{A} = \mu \vec{J} + j\mu\omega\varepsilon \left(-\nabla\phi_e - j\omega\vec{A} \right) \tag{2.19}$$

$$\nabla \left(\nabla \cdot \vec{A} \right) - \nabla^2 \vec{A} = \mu \vec{J} - j\omega\mu\varepsilon\nabla\phi_e + \omega^2\mu\varepsilon\vec{A} \tag{2.20}$$

$$\nabla^2 \vec{A} + k^2 \vec{A} = -\mu \vec{J} + \nabla \left(\nabla \cdot \vec{A} + j\omega\mu\varepsilon\phi \right) \tag{2.21}$$

where $k^2 = \omega^2\mu\varepsilon$ [1]

Equation (2.11) is interesting as it assists in the definition of the divergence of $\vec{A}$ in a similar way to how the curl of $\vec{A}$ was defined in (2.7). Both the divergence and curl of $\vec{A}$ are required to uniquely define the magnetic vector potential $\vec{A}$. By defining the divergence as:

$$\nabla \cdot \vec{A} = -j\omega\mu\varepsilon\phi_e \tag{2.22}$$

Equation (2.21) can be redefined as:

$$\nabla^2 \vec{A} + k^2 \vec{A} = -\mu \vec{J} \tag{2.23}$$

which is the vector wave equation for $\vec{A}$. It is interesting to note that equation (2.22) is called the *Lorentz gauge*, which can be used to define the electric scalar

[1] k is known as the wavenumber or propagation constant and given by $2\pi/\lambda$

potential:

$$\phi_e = -\frac{1}{j\omega\mu\varepsilon}\nabla\cdot\vec{A} \tag{2.24}$$

This allows the electric field defined in equation (2.14) to be provided by :

$$\vec{E}_A = -j\omega\vec{A} - j\frac{1}{j\omega\mu\varepsilon}\nabla\left(\nabla\cdot\vec{A}\right) \tag{2.25}$$

Now once $\vec{A}$ is known, the magnetic field $\vec{H}_A$ created by the magnetic vector potential $\vec{A}$ can be found from (2.8) and the electric field $\vec{E}_A$ can be found from (2.25).

This is only the magnetic and electric field contributions from the magnetic vector potential. In a similar way the electric vector potential $\vec{A}$ can be derived in terms of its magnetic field components $\vec{H}_F$ and $\vec{E}_F$. We start by recognising that the electric flux density $\vec{D}$ is always solenoidal in a source-free region. That is:

$$\nabla\cdot\vec{D} = 0 \tag{2.26}$$

It can therefore be represented by the curl of another vector (which we can arbitrarily create) because it obeys the vector identity:

$$\nabla\cdot\left(-\nabla\times\vec{F}\right) = 0 \tag{2.27}$$

We can therefore define $\vec{D}_F$ by

$$\vec{D}_F = -\nabla\times\vec{F} \tag{2.28}$$

or

$$\vec{E}_F = -\frac{1}{\varepsilon}\nabla\times\vec{F} \tag{2.29}$$

The subscript F denotes that the electric field is due to the electric vector potential $\vec{F}$. This can now be used in Ampère's Law (2.5) remembering that this is a source free region and so $\vec{J} = 0$:

$$\nabla \times \vec{H}_F = j\omega\varepsilon\vec{E}_F \tag{2.30}$$

$$\nabla \times \vec{H}_F = j\omega\varepsilon\left(-\frac{1}{\varepsilon}\nabla \times \vec{F}\right) \tag{2.31}$$

$$\nabla \times \left(\vec{H}_F + j\omega\vec{F}\right) = 0 \tag{2.32}$$

Using the same vector identity from before

$$\nabla \times \left(-\nabla\phi_m\right) = 0 \tag{2.33}$$

provides

$$\vec{H}_F + j\omega\vec{F} = -\nabla\phi_m \tag{2.34}$$

$$\vec{H}_F = -\nabla\phi_m - j\omega\vec{F} \tag{2.35}$$

Here, ϕ_m is an arbitrarily defined magnetic scalar potential that varies with position. We now have $\vec{E}_F$ and $\vec{H}_F$ defined in terms of $\vec{F}$. To find the vector wave equation in terms of $\vec{F}$ we can take the curl of (2.29)

$$\nabla \times \vec{E}_F = \nabla \times \left(-\frac{1}{\varepsilon}\nabla \times \vec{F}\right) \tag{2.36}$$

$$\nabla \times \vec{E}_F = -\frac{1}{\varepsilon}\nabla \times \nabla \times \vec{F} \tag{2.37}$$

$$\nabla \times \vec{E}_F = -\frac{1}{\varepsilon}\left[\nabla\left(\nabla \cdot \vec{F}\right) - \nabla^2\vec{F}\right] \tag{2.38}$$

and so from Faraday's Law (2.4)

$$-\vec{M} - j\omega\mu\vec{H} = -\frac{1}{\varepsilon}\left[\nabla\left(\nabla \cdot \vec{F}\right) - \nabla^2\vec{F}\right] \tag{2.39}$$

$$-\varepsilon\vec{M} + \nabla\left(\nabla \cdot \vec{F}\right) = j\omega\mu\varepsilon\vec{H}_F + \nabla^2\vec{F} \tag{2.40}$$

and using (2.35):

$$-\varepsilon\vec{M} + \nabla\left(\nabla \cdot \vec{F}\right) + j\omega\mu\varepsilon\nabla\phi_m = \omega^2\mu\varepsilon\vec{F} + \nabla^2\vec{F} \tag{2.41}$$

$$\nabla^2\vec{F} + k^2\vec{F} = -\varepsilon\vec{M} + \nabla\left(\nabla\cdot\vec{F} + j\omega\mu\varepsilon\phi_m\right) \tag{2.42}$$

This can be simplified by using a magnetic analogy to the Lorenz gauge:

$$\nabla\cdot\vec{F} = -j\omega\mu\varepsilon\phi_m \tag{2.43}$$

which also provides an expression for the magnetic scalar potential:

$$\phi_m = -\frac{1}{j\omega\mu\varepsilon}\nabla\cdot\vec{F} \tag{2.44}$$

and develops the vector wave equation for the electric vector potential to become

$$\nabla^2\vec{F} + k^2\vec{F} = -\varepsilon\vec{M} \tag{2.45}$$

For completeness the magnetic field can be defined solely in terms of the electric vector potential by using the magnetic scalar potential, (2.44) and equation (2.35) to become:

$$\vec{H}_F = -j\omega\vec{F} - \frac{j}{\omega\mu\varepsilon}\nabla\left(\nabla\cdot\vec{F}\right) \tag{2.46}$$

To summarise up to this point, $\vec{A}$ and $\vec{F}$, once known, now provide solutions to $\vec{E}_A$, $\vec{E}_F$, $\vec{H}_A$ and $\vec{H}_F$ through equations (2.25), (2.29), (2.8) and (2.46) respectively:

$$\vec{E}_A = -j\omega\vec{A} - j\frac{1}{j\omega\mu\varepsilon}\nabla\left(\nabla\cdot\vec{A}\right) \tag{2.47}$$

$$\vec{E}_F = -\frac{1}{\varepsilon}\nabla\times\vec{F} \tag{2.48}$$

$$\vec{H}_A = \frac{1}{\mu}\left(\nabla\times\vec{A}\right) \tag{2.49}$$

$$\vec{H}_F = -j\omega\vec{F} - \frac{j}{\omega\mu\varepsilon}\nabla\left(\nabla\cdot\vec{F}\right) \tag{2.50}$$

The total $\vec{E}$ field and $\vec{H}$ field can now be constructed by taking the vector sum of the two components:

$$\vec{E} = \vec{E}_A + \vec{E}_H = \frac{1}{\varepsilon}\left(\nabla\times\vec{F}\right) - j\omega\left(\vec{A} - \frac{1}{k^2}\nabla\left(\nabla\cdot\vec{A}\right)\right) \tag{2.51}$$

$$\vec{H} = \vec{H}_A + \vec{H}_F = \frac{1}{\mu}\left(\nabla \times \vec{A}\right) - j\omega\left(\vec{F} - \frac{1}{k^2}\nabla\left(\nabla \cdot \vec{F}\right)\right) \tag{2.52}$$

These are the vector expressions that define the $\vec{E}$ and $\vec{H}$ field contributions in a source-free region of space.

2.1.3 Solving the Vector Potentials

Equations have now been defined for the electric and magnetic fields in terms of the vector potentials $\vec{A}$ and $\vec{F}$. If the scattered field can now be defined in terms of $\vec{A}$ and $\vec{F}$ then it can be converted through (2.51) and (2.52) into equations for the electric and magnetic fields. Developing $\vec{A}$ and $\vec{E}$ is not without its challenges however. To simplify the problem, the vector potentials at a point in space are first calculated when the source is at the origin. This will then be extended to the situation when both the measurement point and the scattering source are away from the origin.

2.1.3.1 Source is at the Origin

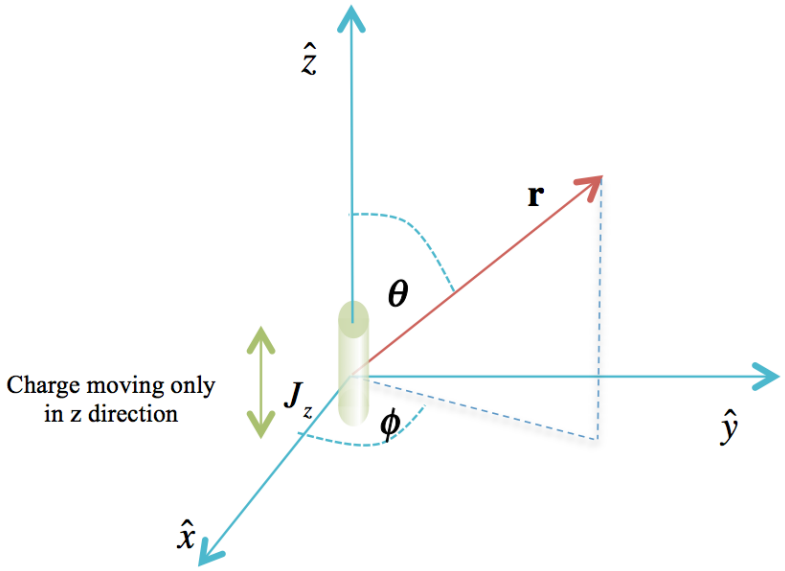


Figure 2.2: An infinitesimal point source is at the origin with current density only in the z direction

Consider Figure 2.2 (with charge centred at origin of x,y,z axes). In this figure there is assumed to be an infinitesimal point source at the origin with current density J_z. Since the current density is directed only along the z-axis then only the z

component of $\vec{A}$ exists and so equation (2.23) can be rewritten as:

$$\nabla^2 A_z + k^2 A_z = -\mu J_z \tag{2.53}$$

At any point removed from the source, $J_z = 0$ and so the wave equation at these locations becomes:

$$\nabla^2 A_z + k^2 A_z = 0 \tag{2.54}$$

In the limiting case the source can actually be considered as a point and so if one considers the spherical coordinate system, (r, θ, ϕ), then A_z will not vary with any angle θ or ϕ. It therefore follows that in the spherical coordinate system $A_z = A_z(r)$. Remembering that the Laplacian operator ∇^2 in spherical coordinates is provided by:

$$\begin{aligned}\nabla^2 f(r,\theta,\phi) = {} & \frac{1}{r^2}\frac{\partial}{\partial r}\left(r^2 \frac{\partial f(r,\theta,\phi)}{\partial r}\right) \\ & + \frac{1}{r^2 \sin\theta}\frac{\partial}{\partial \theta}\left(\sin\theta \frac{\partial f(r,\theta,\phi)}{\partial \theta}\right) + \frac{1}{r^2 \sin^2\theta}\frac{\partial^2 f(r,\theta,\phi)}{\partial \phi^2}\end{aligned} \tag{2.55}$$

then equation (2.54) can be rewritten as:

$$\begin{aligned}\nabla^2 A_z(r) + k^2 A_z(r) = {} & \frac{1}{r^2}\frac{\partial}{\partial r}\left(r^2 \frac{\partial A_z(r)}{\partial r}\right) \\ & + \frac{1}{r^2 \sin\theta}\frac{\partial}{\partial \theta}\left(\sin\theta \frac{\partial A_z(r)}{\partial \theta}\right) + \frac{1}{r^2 \sin^2\theta}\frac{\partial^2 A_z(r)}{\partial \phi^2} + k^2 A_z(r) = 0\end{aligned} \tag{2.56}$$

and so as $A_z(r)$ has no components in θ or ϕ then this becomes

$$\frac{1}{r^2}\frac{\partial}{\partial r}\left(r^2 \frac{\partial A_z(r)}{\partial r}\right) + k^2 A_z(r) = 0 \tag{2.57}$$

Expanding this out using the product rule (and setting the partial derivatives to normal derivatives as the variation is only in *r*), it becomes:

$$\frac{d^2}{dr^2} A_z(r) + \frac{2}{r}\frac{d}{dr} A_z(r) + k^2 A_z(r) = 0 \tag{2.58}$$

This equation is a 2nd order linear homogeneous differential equation. And can be

solved in the standard way (Figure 2.3).

Solving 2nd Order Linear Homogeneous Partial Differential Equations
Suppose we have a function of the form:

$$A\frac{\partial^2}{\partial x^2}f(x)+B\frac{\partial}{\partial x}f(x)+Cf(x)=0$$

Where A, B, and C are all constants. This is called a 2nd order (it contains a second derivative), linear (it is all a function of x) homogeneous (it is equal to zero) differential equation. The objective is to find a solution for y where $y=f(x)$. The first step to solving such equations is to recognise that in order for the terms of the equations to sum to zero (the right hand side), the zeroeth, first and second derivatives of y must all have a common term. For example, this could not occur if $y=x^2$ as the first derivative would be $2x$ and therefore the second derivative would just be 2. In this case the terms would not contain common factors and so could never sum up to zero. To make the equation homogeneous (equal to zero) y must take the form of:

$$y=e^{mx}$$

As derivatives of e^{mx} also contain e^{mx} i.e.:

$$\frac{\partial}{\partial x}e^{mx}=me^{mx}$$
$$\frac{\partial^2}{\partial x^2}e^{mx}=m^2e^{mx}$$

The solution proceeds by substituting the derivatives into the equation that requires solving and then finding solutions to m.

Figure 2.3: How to Solve Linear Homogeneous Partial Differential Equations

To solve this we first recognise that in order for the terms to cancel out and equal zero then the zeroeth, first and second order derivatives must contain the same common factor. And so $A_z(r)$ must be some function of e^x . We start by substituting

$$A_z(r)=e^m r \tag{2.59}$$

where m is some unknown value that we must solve for in order to find $A_z(r)$. The derivatives of (2.59) are

$$\frac{d}{dr}A_z(r)=me^{mr} \tag{2.60}$$

$$\frac{d^2}{dr^2}A_z(r)=m^2e^{mr} \tag{2.61}$$

These can now be substituted into (2.58) to give:

$$e^{mr}\left(m^2+\frac{2}{r}m+k^2\right)=0 \tag{2.62}$$

and as e^{mr} can never equal 0 then

$$\left(m^2+\frac{2}{r}m+k^2\right)=0 \tag{2.63}$$

Solving the quadratic equation in terms of m provides

$$m=\frac{-\frac{2}{r}\pm\sqrt{\frac{4}{r^2}-4k^2}}{2}=-\frac{1}{r}\pm jk\left(1-\frac{1}{k^2r^2}\right)^{1/2} \tag{2.64}$$

The square root term can now be solved using the binomial expansion:

$$\left(1-\frac{1}{(rk)^2}\right)^{1/2}=1+\frac{1}{2}\left(-\frac{1}{(rk)^2}\right)-\frac{1}{8}\left(-\frac{1}{(rk)^2}\right)^2+\frac{1}{16}\left(-\frac{1}{(rk)^2}\right)^3+\ldots \tag{2.65}$$

but as r is large $\frac{1}{r}\to 0$ and so we can ignore higher order terms to give:

$$\left(1-\frac{1}{(rk)^2}\right)^{1/2}=1-\frac{1}{2(rk)^2} \tag{2.66}$$

and so from (2.64)

$$m=-\frac{1}{r}+jk-\frac{j}{2kr^2},\ m=-\frac{1}{r}-jk-\frac{j}{2kr^2} \tag{2.67}$$

Substituting these values for m back into equation (2.59) :

$$\begin{aligned} A_z(r)&=e^{\left(-\frac{1}{r}+jk-\frac{j}{2kr^2}\right)r}=\frac{1}{e}e^{-\frac{j}{2kr}}e^{+jkr}\\ A_z(r)&=e^{\left(-\frac{1}{r}-jk-\frac{j}{2kr^2}\right)r}=\frac{1}{e}e^{-\frac{j}{2kr}}e^{-jkr}\end{aligned} \tag{2.68}$$

This can be simplified further by reforming the centre exponential term as a series

expansion:

$$e^x = \sum_{n=0}^{\infty} \frac{x^n}{n!} = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \frac{x^4}{4!} + \dots \tag{2.69}$$

and so

$$e^{-\frac{j}{2kr}} = 1 - \frac{j}{2kr} + \frac{1}{8k^2r^2} + \frac{j}{48k^3r^3} + \dots \tag{2.70}$$

Putting together (2.68) and (2.70) (and discarding higher order terms of r provides a general solution for $A_z(r)$:

$$A_z(r) = \frac{1}{e}\left(1 - \frac{j}{2kr}\right)e^{+jkr}, \frac{1}{e}\left(1 - \frac{j}{2kr}\right)e^{-jkr} \tag{2.71}$$

$$A_z(r) = C\frac{1}{r}e^{+jkr}, \mathrm{C}\frac{1}{r}e^{-jkr} \tag{2.72}$$

where C is a linear function and comes from recognising that if $A_z(r)$ is a solution to (2.58) then so is $CA_z(r)$. There are two solutions to (2.72), one with $e^{-j\beta r}$ which represents a travelling wave function radiating away from the origin and the second with $e^{+j\beta r}$ which represents a wave function travelling towards the origin. As the wave radiating away from the origin is the one under consideration the appropriate solution for $A_z(r)$ is:

$$A_z(r) = \mathrm{C}\frac{1}{r}e^{-jkr} \tag{2.73}$$

which in the static case, where $\omega = 0$ and therefore $\beta = 0$ becomes:

$$A_z(r) = \frac{C}{r} \tag{2.74}$$

In the presence of the source $J_z \neq 0$ and still considering the static case equation (2.53) becomes

$$\nabla^2 A_z = -\mu J_z \tag{2.75}$$

This can be recognised as Poisson's equation with widely documented solutions. Most commonly Poisson's equation relates the scalar electric potential to the electric charge density :

$$\nabla^2 \phi = -\frac{q}{\varepsilon} \tag{2.76}$$

which has the solution:

$$\phi = \frac{1}{4\pi\varepsilon}\iiint_v \frac{q}{r}dv' \tag{2.77}$$

where r is the distance from any point on the charge density to an observation point and v' represents the volume in the vicinity of the source. Since (2.75) is similar to (2.76) the solution to (2.75) is also similar to (2.77). i.e.:

$$A_z = \frac{\mu}{4\pi}\iiint_v \frac{J_v}{r}dv' \tag{2.78}$$

Equation (2.78) represents the stationary case when the field does not vary with time ($k = 0$) and so to find the time-varying solution we can multiply by the time-varying field equation e^{-jkr} to provide:

$$A_z = \frac{\mu}{4\pi}\iiint_v J_z\frac{e^{-jkr}}{r}dv' \tag{2.79}$$

This is one of the solutions to the vector wave equation (2.53). It follows also that if the current densities were in the x and y directions only then the wave equation could be written as:

$$\nabla^2 A_x + k^2 A_x = -\mu J_x \tag{2.80}$$

or

$$\nabla^2 A_y + k^2 A_y = -\mu J_y \tag{2.81}$$

which also provides the time varying solution in x and y dimension as being:

$$A_x = \frac{\mu}{4\pi}J_x\frac{e^{-jkr}}{r}dv' \tag{2.82}$$

and

$$A_y = \frac{\mu}{4\pi}J_y\frac{e^{-jkr}}{r}dv' \tag{2.83}$$

and so the solution to the vector wave equation in (2.23) is

$$\vec{A} = \frac{\mu}{4\pi}\iiint_v \vec{J}\,\frac{e^{-jkr}}{r}dv' \tag{2.84}$$

which provides the equation for the magnetic vector potential in terms of the volume integration of the electric surface current.

2.1.3.2 Source is Located Away From the Origin

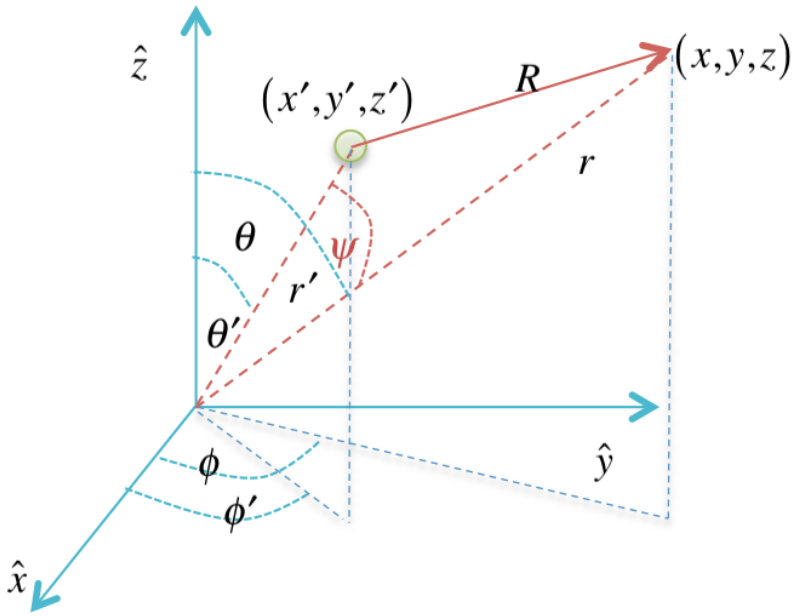


Figure 2.4: Case When Source is Located Away From the Origin

Now consider Figure 2.4, which is the more general case when the source is located at a distance from the origin. Primed coordinates denote the source location (x',y',z'). Equation (2.84) can now be rewritten as:

$$\vec{A}(x,y,z) = \frac{\mu}{4\pi}\iiint_v \vec{J}(x',y',z')\frac{e^{-jkR}}{R}dv' \tag{2.85}$$

where the unprimed coordinates represent the location of the observation point that is a distance R away from the source. The same derivation can now be applied to the vector wave equation in terms of $\vec{F}$ (equation (2.45)) to provide :

$$\vec{F}(x,y,z) = \frac{\varepsilon}{4\pi}\iiint_v \vec{M}(x',y',z')\frac{e^{-jkR}}{R}dv' \tag{2.86}$$

These two equations relate the observed vector potential fields $\vec{A}$ and $\vec{F}$ at the location of an observer to the electric volume density and magnetic volume density within the volume of the source material. If the distribution of the source is assumed to be

homogeneous, these equations can be reformed in terms of only the surface integrals on the surface of the source:

$$\vec{A} = \frac{\mu}{4\pi} \iint_s \vec{J}_s(x',y',z') \frac{e^{-jkR}}{R} ds' \tag{2.87}$$

$$\vec{F} = \frac{\varepsilon}{4\pi} \iint_s \vec{M}_s(x',y',z') \frac{e^{-jkR}}{R} ds' \tag{2.88}$$

Equally the surface integrals can now be formulated in terms of line integrals in which case they become surface electric and magnetic *currents* of the form:

$$\vec{A} = \frac{\mu}{4\pi} \int_C \vec{I}_e(x',y',z') \frac{e^{-jkR}}{R} dl' \tag{2.89}$$

$$\vec{F} = \frac{\varepsilon}{4\pi} \int_C \vec{I}_m(x',y',z') \frac{e^{-jkR}}{R} dl' \tag{2.90}$$

where $\vec{I}_e$ and $\vec{I}_m$ are the electric and magnetic currents respectively on the surface of the source. The vector potentials $\vec{A}$ and $\vec{F}$ have now been defined in terms of the surface currents on a scattering body which are measurable quantities.

To summarise to this point before proceeding to the next section, the fields radiated by surface current densities $\vec{J}_s$ and $\vec{M}_s$ can be found from equations (2.47)-2.50 :

$$\vec{E}_A = -j\omega\vec{A} - j\frac{1}{\omega\varepsilon\mu} \nabla\left(\nabla \cdot \vec{A}\right)$$

$$\vec{E}_F = \frac{1}{\varepsilon}\left(\nabla \times \vec{F}\right)$$

$$\vec{H}_A = \frac{1}{\mu}\left(\nabla \times \vec{A}\right)$$

$$\vec{H}_F = -j\omega\vec{F} - \frac{j}{\omega\mu\varepsilon} \nabla\left(\nabla \cdot \vec{F}\right)$$

where $\vec{A}$ and $\vec{F}$ can be found from the electric and magnetic surface currents flowing over the scattering surface using :

$$\vec{A} = \frac{\mu}{4\pi} \iint_s \vec{J}_s(x',y',z') \frac{e^{-jkR}}{R} ds' \tag{2.91}$$

$$\vec{F} = \frac{\varepsilon}{4\pi} \iint_s \vec{M}_s(x', y', z') \frac{e^{-jkR}}{R} ds' \tag{2.92}$$

2.1.4 Simplification Using the Far-Field Approximation

Equations (2.91) and (2.92) form valid solutions for all observation points but involve an integration over the entire space occupied by $\vec{J}$ and $\vec{M}$. The integration is very difficult to solve in general terms because of the variation in R over the surface and its existence in the exponential term. Fortunately this can be simplified by considering only the situation where the observer is in the far field and is the next topic in the formulation. First consider the equation for the range R between the surface and the

How Valid is the Far-Field Approximation?

The range from a source to an observer (using the angles defined in Figure 2.4 is given by:

$$R = \left(r^2 + r'^2 - 2rr' \cos\psi_s\right)^{1/2}$$

Which can be approximated through the binomial expansion as

$$R = r - r' \cos\psi_s + \frac{1}{r}\left(\frac{r'^2}{2}\sin^2\psi_s\right) + \frac{1}{r^2}\left(\frac{r'^3}{2}\cos\psi_s \sin^2\psi_s\right) + \ldots$$

When it is assumed that r is large then higher order terms (r^2 and above) approach zero and so the far-field approximation can be written using only the first 3 terms.

$$R \approx r - r' \cos\psi_s + \frac{1}{r}\left(\frac{r'^2}{2}\sin^2\psi_s\right)$$

But what is the range of validity of this equation? This can be defined by looking at the sensitivity of the third term in the equation. The third term takes a maximum value when $\psi = \pi/2$ and becomes

$$\frac{1}{r}\left(\frac{r'^2}{2}\right)$$

It has been shown in many examples [78] that a maximum phase error of $\pi/8$ radians will have an insignificant effect on any analytical formulations and so we can set the maximum tolerance of this equation to be

$$\frac{2\pi}{\lambda}\frac{r'^2}{2r} \leq \frac{\pi}{8}$$

Which provides the definition of the Fraunhofer zone for transition from Near field to far field as

$$r \geq \frac{2D^2}{\lambda},\ D = 2r'$$

Figure 2.5: Validity of the Far-Field Approximation

observer using the cosine rule and Figure 2.4:

$$R = \left(r^2 + r'^2 - 2rr'\cos\psi_s\right)^{1/2} \tag{2.93}$$

where ψ_s is the angle between the line connecting the scatterer to the origin and the line connecting the origin and the observation point. Using the binomial expansion we can rewrite equation (2.93) as

$$R = r - r'\cos\psi_s + \frac{1}{r}\left(\frac{r'^2}{2}\sin^2\psi_s\right) + \frac{1}{r^2}\left(\frac{r'^3}{2}\cos\varphi\sin^2\psi_s\right) + \ldots \tag{2.94}$$

If it is assumed that r is large, higher powers of r, $1/r^n = 0$ for $n = 2,3,4\ldots$ can be ignored. The equation then approximates to:

$$R \approx r - r'\cos\psi_s + \frac{1}{r}\left(\frac{r'^2}{2}\sin^2\psi_s\right) \tag{2.95}$$

This far-field approximation, Figure 2.5, provides the means to safely remove the third term in this equation providing that the observation point is at a distance of

$$r \geq \frac{2D^2}{\lambda} \tag{2.96}$$

away from the scattering body where D is the diameter of a scattering body ($D = 2r'$). Equation (2.94) now becomes:

$$R \approx r - r'\cos\psi_s \tag{2.97}$$

The usual implementation of these simplifications into equations (2.91) and (2.92) is to approximate the R in the phase term ($e^{-j\beta R}$) by equation (2.97) and to replace the R in the denominator by $R \simeq r$. Collectively these simplifications are called the far-field approximation in the literature:

The Far Field Approximation

$$\left.\begin{array}{ll} R \simeq r - r' \cos\psi_s & \text{for phase terms} \\ R \simeq r & \text{for amplitude terms} \end{array}\right\} \tag{2.98}$$

These approximations can now be substituted into the equations for $\vec{A}$ and $\vec{F}$. Assuming that the current densities are on the surface of the scattering object equation (2.91) can be rewritten as

$$\vec{A} = \frac{\mu}{4\pi} \iint_s \vec{J}_s \frac{e^{-jkR}}{R} ds' \approx \frac{\mu}{4\pi} \frac{e^{-jkr}}{r} \iint_s \vec{J}_s e^{jkr' \cos\psi_s} ds' \tag{2.99}$$

$$\vec{A} \approx \frac{\mu e^{-jkr}}{4\pi r} N \tag{2.100}$$

where N is

$$N = \iint_s \vec{J}_s e^{jkr' \cos\psi_s} ds' \tag{2.101}$$

Equally for $\vec{F}$:

$$\vec{F} = \frac{\varepsilon}{4\pi} \iint_s \vec{M}_s \frac{e^{-jkR}}{R} ds' \approx \frac{\varepsilon}{4\pi} \frac{e^{-jkr}}{r} \iint_s \vec{M}_s e^{jkr' \cos\psi_s} ds' \tag{2.102}$$

$$\vec{F} \approx \frac{\varepsilon e^{-jkr}}{4\pi r} L \tag{2.103}$$

where L is

$$L = \iint_s \vec{M}_s e^{jkr' \cos\psi_s} ds' \tag{2.104}$$

2.1.5 Rebuilding the E and H Fields From the Vector Potentials

It now remains to equate the vector potentials $\vec{A}$ and $\vec{F}$ into equations for the electric and magnetic fields, $\vec{E}$ and $\vec{H}$. This is achieved by using equations (2.47) through (2.50).

The analysis starts with:

$$\vec{E}_A = -j\omega\vec{A} - j\frac{1}{\omega\varepsilon\mu}\nabla\left(\nabla\cdot\vec{A}\right) \tag{2.105}$$

which can be rewritten as

$$\vec{E}_A = -j\omega\left(\vec{A} + \frac{1}{k^2}\nabla\left(\nabla\cdot\vec{A}\right)\right) \tag{2.106}$$

Considering $\vec{A}$ in terms of its spherical coordinates:

$$\vec{A} = \hat{a}_r A_r(r,\theta,\phi) + \hat{a}_\theta A_\theta(r,\theta,\phi) + \hat{a}_\phi A_\phi(r,\theta,\phi) \tag{2.107}$$

and remembering that the formula for grad and divergence in spherical coordinates is

$$\nabla g = \frac{\partial g}{\partial r}\hat{r} + \frac{1}{r}\frac{\partial g}{\partial\theta}\hat{\theta} + \frac{1}{r\sin\theta}\frac{\partial g}{\partial\phi}\hat{\phi} \tag{2.108}$$

$$\nabla\cdot\vec{G} = \frac{1}{r^2}\frac{\partial(r^2G_r)}{\partial r} + \frac{1}{r\sin\theta}\frac{\partial}{\partial\theta}(G_\theta\sin\theta) + \frac{1}{r\sin\theta}\frac{\partial}{\partial\phi}G_\phi \tag{2.109}$$

and so

$$\nabla\cdot\vec{A} = \frac{1}{r}\left(\frac{1}{r}\frac{\partial(r^2A_r)}{\partial r} + \frac{1}{\sin\theta}\frac{\partial}{\partial\theta}(A_\theta\sin\theta) + \frac{1}{\sin\theta}\frac{\partial}{\partial\phi}A_\phi\right) \tag{2.110}$$

which leads to

$$\begin{aligned}\nabla\left(\nabla\cdot\vec{A}\right) = {} & \frac{\partial}{\partial r}\left(\frac{1}{r}\left(\frac{1}{r}\frac{\partial(r^2A_r)}{\partial r} + \frac{1}{\sin\theta}\frac{\partial}{\partial\theta}(A_\theta\sin\theta) + \frac{1}{\sin\theta}\frac{\partial}{\partial\phi}A_\phi\right)\right)\hat{r} \\ & + \frac{1}{r^2}\frac{\partial}{\partial\theta}\left(\frac{1}{r}\frac{\partial(r^2A_r)}{\partial r} + \frac{1}{\sin\theta}\frac{\partial}{\partial\theta}(A_\theta\sin\theta) + \frac{1}{\sin\theta}\frac{\partial}{\partial\phi}A_\phi\right)\hat{\theta} \\ & + \frac{1}{r^2\sin\theta}\frac{\partial}{\partial\phi}\left(\frac{1}{r}\frac{\partial(r^2A_r)}{\partial r} + \frac{1}{\sin\theta}\frac{\partial}{\partial\theta}(A_\theta\sin\theta) + \frac{1}{\sin\theta}\frac{\partial}{\partial\phi}A_\phi\right)\hat{\phi}\end{aligned} \tag{2.111}$$

Consider now only the radial terms of the equation:

$$\frac{\partial}{\partial r}\left(\frac{1}{r}\left(\frac{1}{r}\frac{\partial(r^2A_r)}{\partial r} + \frac{1}{\sin\theta}\frac{\partial}{\partial\theta}(A_\theta\sin\theta) + \frac{1}{\sin\theta}\frac{\partial}{\partial\phi}A_\phi\right)\right)\hat{r} \tag{2.112}$$

Differentiating this provides:

$$\left(\frac{\partial^2}{\partial r^2}A_r+\frac{1}{r^2}\left(2\left(r\frac{\partial}{\partial r}A_r+\frac{A_r}{r^2}\right)-\frac{1}{\sin\theta}\frac{\partial}{\partial\theta}\left(A_\theta\sin\theta\right)-\frac{1}{\sin\theta}\frac{\partial}{\partial\phi}A_\phi\right)\right)\hat{r} \tag{2.113}$$

If one assumes that A_r does not vary wildly with r ([79] pp280) then the remaining parts of this equation are all related to $1/r^2$. If only the far field is considered ($1/r^n = 0$, with $(n = 2,3,\ldots)$) then equation (2.111) can be considered as

$$\nabla\left(\nabla\cdot\vec{A}\right)\approx 0 \text{ (far field)} \tag{2.114}$$

and so

$$\left.\begin{array}{c}(E_A)_r\simeq 0\\(E_A)_\theta\simeq -j\omega A_\theta\\(E_A)_\phi\simeq -j\omega A_\phi\end{array}\right]\rightarrow \vec{E}_A=-j\omega\vec{A}\text{ (for }\theta\text{ and }\phi\text{ components only)} \tag{2.115}$$

The same arguments can also be made for the $\vec{F}$ component of the $\vec{H}$ field in equation (2.50) to provide:

$$\left.\begin{array}{c}(H_F)_r\simeq 0\\(H_F)_\theta\simeq -j\omega F_\theta\\(H_F)_\phi\simeq -j\omega F_\phi\end{array}\right]\rightarrow \vec{H}_F\simeq -j\omega\vec{F}\text{ (for }\theta\text{ and }\phi\text{ components only)} \tag{2.116}$$

As the $\vec{E}$ and $\vec{H}$ fields are orthogonal to each other and are related through the intrinsic impedance of the material by:

$$(E_F)_\theta\simeq +Z_0(H_F)_\phi=-j\omega Z_0F_\phi \tag{2.117}$$

$$(E_F)_\phi\simeq -Z_0(H_F)_\theta=+j\omega Z_0F_\theta \tag{2.118}$$

$$(H_A)_\theta\simeq -\frac{(E_A)_\phi}{Z_0}=+j\omega\frac{A_\phi}{Z_0} \tag{2.119}$$

$$(H_A)_\phi\simeq +\frac{(E_A)_\theta}{Z_0}=-j\omega\frac{A_\theta}{Z_0} \tag{2.120}$$

The vector potential components of the fields can now be combined to give:

$$E_r \simeq 0 \tag{2.121}$$

$$E_\theta \simeq (E_A)_\theta + (E_F)_\theta = -j\omega\left(A_\theta - Z_0 F_\phi\right) \tag{2.122}$$

$$E_\phi \simeq (E_A)_\phi + (E_F)_\phi = -j\omega\left(A_\phi - Z_0 F_\theta\right) \tag{2.123}$$

$$H_r \simeq 0 \tag{2.124}$$

$$H_\theta \simeq (H_A)_\theta + (H_F)_\theta = +\frac{j\omega}{Z_0}\left(A_\phi - Z_0 F_\theta\right) \tag{2.125}$$

$$H_\phi \simeq (H_A)_\phi + (H_F)_\phi = -\frac{j\omega}{Z_0}\left(A_\theta + Z_0 F_\phi\right) \tag{2.126}$$

And so using $A_\theta, A_\phi, F_\theta$ and F_ϕ from equations (2.100), (2.101), (2.103) and (2.104) i.e.:

$$A_\theta = \frac{\mu e^{-jkr}}{4\pi r} N_\theta \tag{2.127}$$

$$A_\phi = \frac{\mu e^{-jkr}}{4\pi r} N_\phi \tag{2.128}$$

$$F_\theta = \frac{\varepsilon e^{-jkr}}{4\pi r} L_\theta \tag{2.129}$$

$$F_\phi = \frac{\varepsilon e^{-jkr}}{4\pi r} L_\phi \tag{2.130}$$

equations (2.121) to (2.126) can be reduced to become

$$E_r \simeq 0 \tag{2.131}$$

$$E_\theta \simeq -\frac{jke^{-jkr}}{4\pi r}\left(L_\phi + Z_0 N_\theta\right) \tag{2.132}$$

$$E_\phi \simeq +\frac{jke^{-jkr}}{4\pi r}\left(L_\theta - Z_0 N_\phi\right) \tag{2.133}$$

$$H_r \simeq 0 \tag{2.134}$$

$$H_\theta \simeq +\frac{jke^{-jkr}}{4\pi r}\left(N_\phi - \frac{L_\theta}{Z_0}\right) \tag{2.135}$$

$$H_\phi \simeq -\frac{jke^{-jkr}}{4\pi r}\left(N_\theta + \frac{L_\phi}{Z_0}\right) \tag{2.136}$$

The final EM fields caused by electric and magnetic surface currents (assuming the far-field approximation), can therefore be determined using equations (2.101) and (2.104) for the functions L and N. The easiest way to calculate the spherical components of L and N is to first consider the current densities in a convenient coordinate system. Consider the situation where the electric and magnetic current densities are specified in the Cartesian coordinate system:

$$N = \iint_s \vec{J}_s e^{jkr'\cos\psi_s} ds' = \iint_s \left(\hat{a}_x J_x + \hat{a}_y J_y + \hat{a}_z J_z\right) e^{jkr'\cos\psi_s} ds' \tag{2.137}$$

$$L = \iint_s \vec{M}_s e^{jkr'\cos\psi_s} ds' = \iint_s \left(\hat{a}_x M_x + \hat{a}_y M_y + \hat{a}_z M_z\right) e^{jkr'\cos\psi_s} ds' \tag{2.138}$$

The Cartesian components of the current densities can be converted into spherical components using the usual component transformation:

$$\begin{bmatrix} \hat{a}_x \\ \hat{a}_y \\ \hat{a}_z \end{bmatrix} = \begin{bmatrix} \sin\theta\cos\phi & \cos\theta\cos\phi & -\sin\phi \\ \sin\theta\sin\phi & \cos\theta\sin\phi & \cos\phi \\ \cos\theta & -\sin\theta & 0 \end{bmatrix} \begin{bmatrix} \hat{a}_r \\ \hat{a}_\theta \\ \hat{a}_\phi \end{bmatrix} \tag{2.139}$$

Replacing Cartesian components with spherical components in (2.137) and (2.138):

$$N = \iint_s \begin{pmatrix} \left(J_x\sin\theta\cos\phi + J_y\sin\theta\sin\phi + J_z\cos\theta\right)\hat{a}_r \\ + \left(J_x\cos\theta\cos\phi + J_y\cos\theta\sin\phi - J_z\sin\theta\right)\hat{a}_\theta \\ + \left(-J_x\sin\phi + J_y\cos\phi\right)\hat{a}_\phi \end{pmatrix} e^{jkr'\cos\psi_s} ds' \tag{2.140}$$

which provides:

$$N_\theta = \iint_s \left(J_x\cos\theta\cos\phi + J_y\cos\theta\sin\phi - J_z\sin\theta\right) e^{jkr'\cos\psi_s} ds' \tag{2.141}$$

and

$$N_\phi = \iint_s \left(-J_x\sin\phi + J_y\cos\phi\right) e^{jkr'\cos\psi_s} ds' \tag{2.142}$$

and equally for L:

$$L_\theta = \iint_s \left(M_x cod\theta \cos\phi + M_y \cos\theta \sin\phi - M_z \sin\theta \right) e^{jkr' \cos\psi_s} ds' \tag{2.143}$$

and

$$L_\phi = \iint_s \left(-M_x \sin\phi + M_y \cos\phi \right) e^{jkr' \cos\psi_s} ds' \tag{2.144}$$

Finally the scattered fields at any far-field location can be derived once the current densities on the surface of a scattering body are known. The next section will show how these solutions can be applied to scattering primitives commonly used for radar EM field calculations by developing a technique for calculating the surface currents for an inbound electromagnetic wave.

2.2 Electromagnetic Material Properties

The previous section described an approach that allows the electric and magnetic fields to be calculated in the far-field away from a scattering body. It was shown that the solution depended on the electric and magnetic currents on the surface of the scattering body:

$$E_r \simeq 0 \tag{2.145}$$

$$E_\theta \simeq -jk \frac{e^{-jkr}}{4\pi r} \left(L_\phi + Z_0 N_\theta \right) \tag{2.146}$$

$$E_\phi \simeq +\frac{jke^{-jkr}}{4\pi r} \left(L_\theta - Z_0 N_\phi \right) \tag{2.147}$$

$$H_r \simeq 0 \tag{2.148}$$

$$H_\theta \simeq +\frac{jke^{-jkr}}{4\pi r} \left(N_\phi - \frac{L_\theta}{Z_0} \right) \tag{2.149}$$

$$H_\phi \simeq -\frac{jke^{-jkr}}{4\pi r} \left(N_\theta + \frac{L_\phi}{Z_0} \right) \tag{2.150}$$

with

$$N_\theta = \iint_s \left(J_x \cos\theta \cos\phi + J_y \cos\theta \sin\phi - J_z \sin\theta \right) e^{jkr' \cos\psi_s} ds' \tag{2.151}$$

$$N_\phi = \iint_s \left(-J_x \sin\phi + J_y \cos\phi\right) e^{jkr' \cos\psi_s} ds' \tag{2.152}$$

$$L_\theta = \iint_s \left(M_x cod\theta \cos\phi + M_y \cos\theta \sin\phi - M_z \sin\theta\right) e^{jkr' \cos\psi_s} ds' \tag{2.153}$$

$$L_\phi = \iint_s \left(-M_x \sin\phi + M_y \cos\phi\right) e^{jkr' \cos\psi_s} ds' \tag{2.154}$$

The overall aim is to calculate the fields at a receiver caused by an incident EM wave on a scattering body. The above development is therefore only half of the final analysis. It is equally important to determine how those surface currents arose from an incident EM wave. This is the subject of this section. After the equations are developed there will then be a discussion on the implementation of this formulation on two simple scattering primitives: the rectangular patch and the triangle. It will be shown that the induced surface currents are a function of the intrinsic material properties.

A common approach to EM scattering problems assumes that EM reflecting bodies within a scene are composed of facets that do not contain source charges or support the generation of electric (or magnetic) fields. This limits simulations to perfect electrical conductors (PECs). The intention of this work however is to develop simulated data that is representative of a Synthetic Aperture Radar scene which requires a more general solution using realistic material properties. In order to base the subsequent derivations on a solid foundation, the section will first discuss how material properties determine the flow (or resistance to flow) of charge within a medium. These will more easily facilitate the understanding of the boundary conditions on a material surface that do not rely on the PEC case.

The formulation then describes an incident ray in terms of orthogonal polarisation vectors. This is because it is easier to consider orthogonal polarisations individually and then to combine them afterwards to find the solution of any arbitrary polarisation.

The approach taken is adapted from the one used by Nikolaos Faros [86] but here the work has been extended to provide the magnetic surface current density equations required for accurate modelling of a wider range of materials. The overall

aim of the derivation is to arrive at a set of equations that describe the surface currents on a facet induced by an incident EM wave. These currents can then be plugged into the derivation in the previous chapter for the scattered fields at a radar receiver. Due to the complexity of the derivation a figure is provided here (Figure 2.6) to provide an overview of the approach.

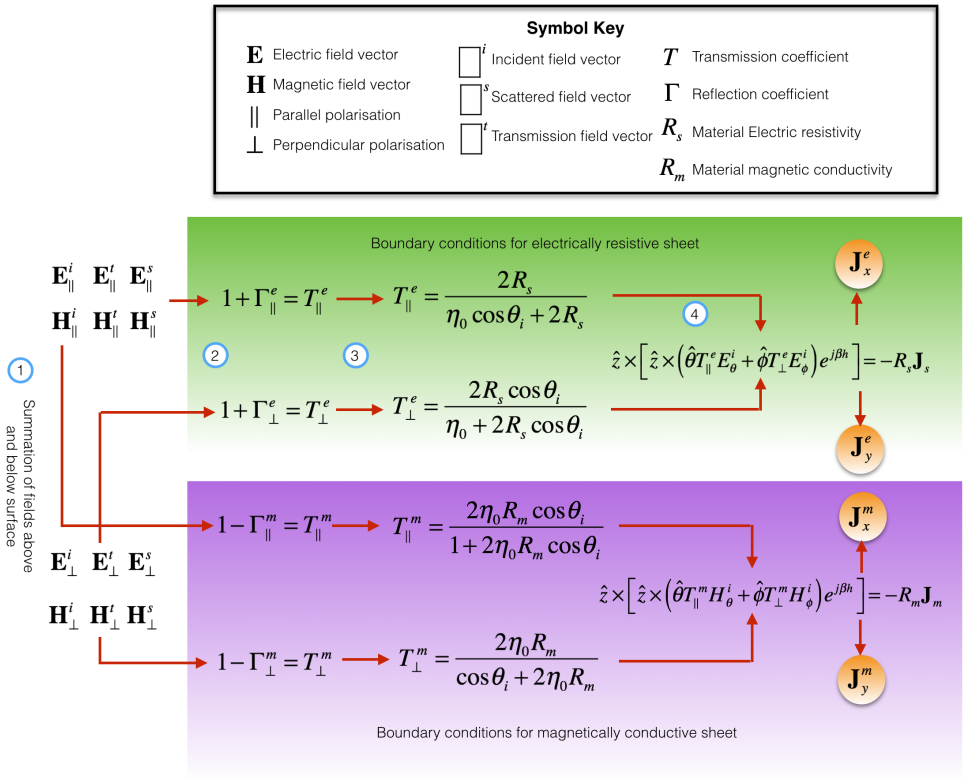


Figure 2.6: Steps to Determine Surface Currents From an Incident EM Field

The figure comprises the key steps required to determine the surface currents by way of highlighting the essential equations required to progress from one step to the next.

The lower coloured box represents all the derivations that require boundary conditions that are unique to the magnetically conductive sheet whilst the upper coloured box represents the derivations that require the boundary conditions for the electrically resistive sheet. The section therefore starts by presenting a careful consideration of the boundary conditions for a combination of magnetic and electric currents.

The analysis starts on the left (1) with the formulation of the fields both above and below the illuminated sheet but considering the parallel ($\parallel$) and perpendicular ($\perp$) polarisations individually. These components are then used to derive the relationships between the transmission coefficients and the reflection coefficients for both polarisation cases and for the electric and magnetic sheet cases (2). This information is then used (3) together with the boundary conditions and the field components from (1) to derive a set of equations that describe reflection and transmission coefficients in terms of the actual material properties and angle of incidence of the inbound ray. The material properties at this stage are represented as either an electrical resistivity (sheet resistivity) R_s or a magnetic conductivity R_m. A discussion of these properties will be presented.

Once these equations have been derived, the relevant boundary conditions and the starting geometry of the fields can be used to derive the final equations that specify the electric surface current components J_x^e, J_y^e and the magnetic surface current J_x^m, J_y^m (4). These are equivalent to M_x, M_y in equations (2.153) and (2.154).

The derivation starts with a discussion on material properties.

2.2.1 Electrical and Magnetic Material Properties

Before boundary conditions can be considered, it is worth having a brief recap on how the properties of a material affect the flow of charge within it.

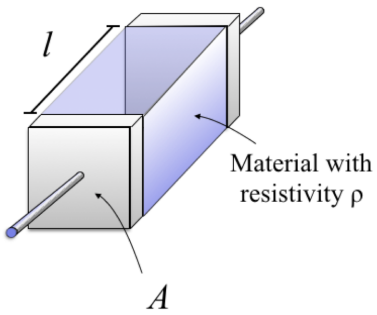


Figure 2.7: Resistance Vs Resistivity

The resistance of a material is a function of the material's ability to transport charge, together with the cross sectional area of the material and its length (Figure 2.7). A longer length of material will have a higher resistance than a shorter

length of material of the same cross sectional area and equally a piece of material that has a larger cross section than a similar piece with the same resistivity, will allow more charge to flow through it. It therefore follows that the definition of resistance is

$$R = \rho \frac{l}{A} \tag{2.155}$$

where R is the resistance of the material, l is the length and A is the cross sectional area of the material. The parameter ρ is called the resistivity and is an inherent property of the material. It's SI units are the Ohm metre.

The conductivity of a material is a measure of the amount of energy an electric field loses as it passes through the medium. Its units are Siemens per metre and it is the reciprocal of resistivity (measured in Ohm metres). When a material has zero conductivity ($\rho \to \infty$) then no electric charge can flow through it. As a result any electric field over the material is unaffected by it and so it is electromagnetically transparent to any incident EM wave. Good examples of a lossless medium are air and vacuum.

A material with a non-zero, but finite conductivity is called a lossy medium. In this case electric charge can flow as current (or current per unit area also known as electric charge density $\vec{J}$). When an EM wave is incident on such a medium it causes current to flow within the material. This motion of charge results in resistive heating which attenuates the incident field (hence lossy).

The amount of current flowing through the medium is a function of the conductivity and the supported electric field and is given by Ohm's Law:

$$\vec{J} = \sigma \vec{E} \tag{2.156}$$

The final category of conductivity occurs with materials of very high conductivity ($\rho \to 0$) called conductors. In such cases when an EM field is incident on the material, charge can flow very easily. This quickly reduces the instantaneous electric field. If no E-field can be supported in the medium then again the material is lossless.

The extreme case of high electrical conductivity occurs in a hypothetical medium called a Perfect Electrical Conductor (PEC). In such a material the conductivity is infinite (zero resistance). According to equation (2.156) this would result in an infinite current density, which clearly could not occur, unless the electric field was zero.

Material	σ [S/m]	Category
Vacuum	0	Lossless
Teflon	$\sim 10^{-24}$	Lossless
Air	$\sim 10^{-15}$	Lossless
Rubber	$\sim 10^{-14}$	Lossless
Glass	$\sim 10^{-12}$	Lossless
Wood (damp)	$\sim 10^{-4}$	Lossy
Silicon	$\sim 1.56\times10^{-3}$	Lossy
Germanium	2.17	Lossy
Sea Water	4.8	Lossy
Carbon	2.00×10^{3}	Lossy
Mercury	1.02×10^{6}	Conductor
Iron	1×10^{7}	Conductor
Nickel	1.43×10^{7}	Conductor
Tungsten	1.79×10^{7}	Conductor
Aluminium	3.50×10^{7}	Conductor
Copper	5.96×10^{7}	Conductor
Silver	6.30×10^{7}	Conductor

Table 2.2: Conductivity of Selected Materials

2.2.2 Boundary Conditions

Consider Figure 2.8, which shows a perfectly conducting medium, illuminated with a planar electromagnetic wave

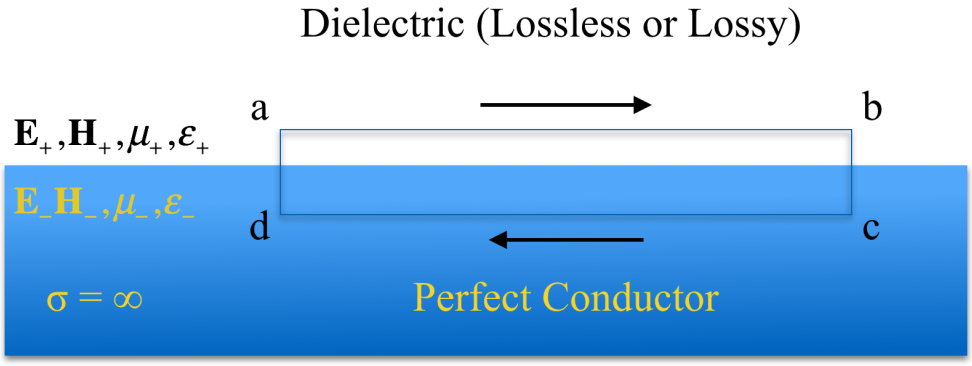


Figure 2.8: Boundary of a Perfectly Conducting Medium

From the integral form of Faraday's Law:

$$\oint_s \vec{E} \cdot d\vec{l} = -\frac{d}{dt} \iint_s \vec{B} \cdot d\vec{s} \tag{2.157}$$

Considering an arbitrary rectangular loop (Figure 2.8) that encloses both mediums, this can be rewritten as:

$$\int_a^b \vec{E}_+ \cdot d\vec{l} + \int_b^c \vec{E}_\pm \cdot d\vec{l} + \int_c^d \vec{E}_- \cdot d\vec{l} + \int_d^a \vec{E}_\pm \cdot d\vec{l} = -\frac{d}{dt} \int_{abcd} \vec{B} \cdot d\vec{s} \tag{2.158}$$

If the rectangle has only a very small area such that it only considers the very small transition point between medium one and medium two then the lengths $\overrightarrow{bc}$ and $\overrightarrow{da}$ can be ignored as having no contribution to the left hand side. This provides:

$$\int_b^c \vec{E}_\pm \cdot d\vec{l} = \int_d^a \vec{E}_\pm \cdot d\vec{l} \to 0 \tag{2.159}$$

Also, as material two is a perfect conductor, it cannot support any electric field. Any potential difference within the medium is immediately reduced to zero through the (hypothetical) infinitely quick redistribution of charge. This leaves equation (2.158) as:

$$\int_a^b \vec{E}_+ \cdot d\vec{l} = -\frac{d}{dt} \int_{abcd} \vec{B} \cdot d\vec{s} \tag{2.160}$$

On the right hand side of this equation, the area enclosed within the loop is now infinitely small and so:

$$-\frac{d}{dt} \int_{abcd} \vec{B} \cdot d\vec{s} \to 0 \tag{2.161}$$

leaving :

$$\int_a^b \vec{E}_+ \cdot d\vec{l} = 0 \tag{2.162}$$

This is an interesting equation. It is known that $\vec{E}_+$ is not zero. It is the electric field vector for the EM wave propagating through medium one. To make the summation of $\vec{E}_+$ over just the region from a to b equal to zero then consider that $\overrightarrow{ab}$ is a small

but finite distance. The equation then becomes:

$$\vec{E}_t \cdot \Delta = 0 \tag{2.163}$$

where $\vec{E}_t$ is the component of the electric field that is tangential to the surface boundary. i.e.

$$\vec{E}_t = \vec{E}_+ \times \hat{n} \tag{2.164}$$

where $\hat{n}$ is the surface normal. There are two interesting consequences of (2.162). First, consider that as there is no electric field between any two points on the surface, a perfect electrical conductor is an equipotential surface. The second is that any electric field becomes completely orthogonal to the surface. That is the electric field lines are all aligned with the surface normal. The strength of the perpendicular electric field lines can be quantified by looking at Figure 2.9.

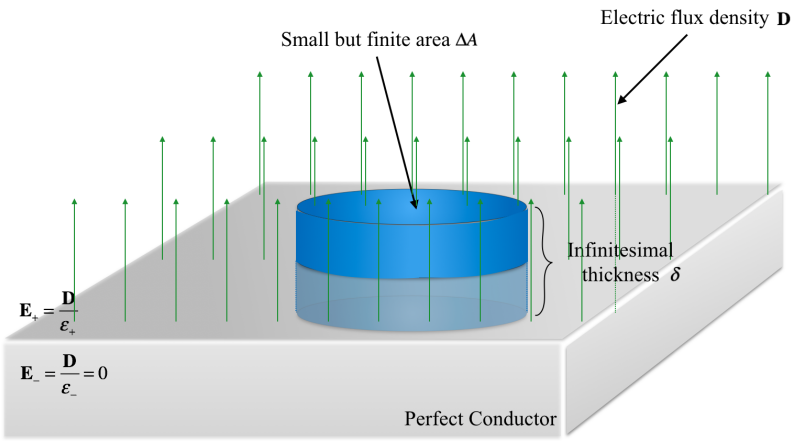


Figure 2.9: Electric Field Lines at the Boundary of a Perfect Electrical Conductor

This figure contains a pillbox shape that sits on the boundary between the two media. The electric flux density through the pillbox is constant and the electric permittivity ε_- in the perfect conductor is zero then the E field $\vec{E}_-$ is also zero as described above.

From the integral form of Gauss' Law:

$$\oiint_S \vec{D} \cdot d\vec{s} = \iiint_V \rho_e dv \tag{2.165}$$

If there is a volume charge density then the right hand side of the equation could be set to zero as the volume is infinitesimally small. There is however a surface charge density on the very surface of the conductor. As a result, the right hand side of the equation becomes:

$$\iint_s \rho_e d\vec{s} = \Delta A \cdot \rho_e \tag{2.166}$$

The left hand side of the equation represents the integral over the surface of the pillbox of the electric flux density flowing through it and can be consider as three individual contributions: the top; the bottom and the sides.

The left hand side can therefore be written as:

$$\iint_s \vec{D} \cdot d\vec{s} = \int_{\text{side}} \vec{D} \cdot d\vec{s} + \int_{\text{top}} \vec{D} \cdot d\vec{s} + \int_{\text{bottom}} \vec{D} \cdot d\vec{s} \tag{2.167}$$

The integral of $\vec{D} \cdot d\vec{s}$ over the side is zero as the height of the pillbox is very small and, as discussed, the direction of the fields at the boundary are in the same direction as the surface normal. The integral over the bottom surface must be zero as there are no fields supported inside a perfect conductor. As a result equation (2.167) becomes:

$$\oiint_s \vec{D} \cdot d\vec{s} = \int_{\text{top}} \vec{D} \cdot d\vec{s} = \Delta A \vec{D} \cdot \hat{n} \tag{2.168}$$

Combining (2.166 and (2.168) into (2.165)):

$$\Delta A \vec{D} \cdot \hat{n} = \Delta A \cdot \rho_e \tag{2.169}$$

$$\vec{D} \cdot \hat{n} = \rho_e \tag{2.170}$$

The boundary conditions of the electric field for a perfect conductor have now been derived.

What about the magnetic fields? The arguments are very similar. The analysis starts with Ampère's Law and Figure 2.10

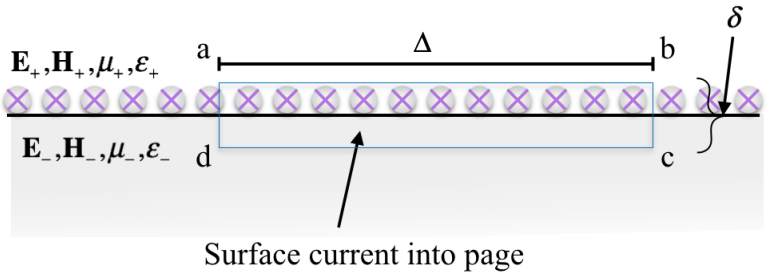


Figure 2.10: Magnetic Field Boundary Conditions for Perfect Electrical Conductor

Ampère's Law in integral form is:

$$\oint_{abcd} \vec{H}\cdot d\vec{l} = \int_{abcd} \vec{J}_s \cdot d\vec{s} + \frac{d}{dt}\iint_{abcd} \vec{D}\cdot d\vec{s} \tag{2.171}$$

In the figure there is a surface current density at the boundary between the two media flowing into the page. The field that drives this surface current must come from medium 1 as it has already been seen that medium 2 cannot support any fields. It has also been shown that the tangential electric field is zero. As a result, if the thickness of the integral loop is reduced then eventually $\vec{D}\cdot d\vec{s} \to 0$. The first term on the right hand side however does not shrink to zero. The reason for this is that the surface current can be infinite for a PEC (as the electric field approaches zero). The left hand side of equation (2.171) can be solved using the same reasoning for the electric field. Namely:

$$\int_a^b \vec{H}_+ \cdot d\vec{l} = H_t \Delta \tag{2.172}$$

and so

$$H_t \Delta = J_s \Delta \tag{2.173}$$

or

$$\hat{n} \times \vec{H}_+ = \vec{J}_s \tag{2.174}$$

The normal component of the magnetic field can be derived in the same way as the

electric field using Figure 5 4 and from

$$\oiint \vec{B} \cdot ds = 0 \tag{2.175}$$

As the right hand side must be zero (there are no magnetic monopoles) then the magnetic flux through the pillbox must also be zero.

The electromagnetic boundary conditions have now been derived for the surface of a perfect electrical conductor:

$$\hat{n} \times \vec{E}_+ = 0 \tag{2.176}$$

$$\hat{n} \cdot \vec{D}_+ = \rho_e \tag{2.177}$$

$$\hat{n} \times \vec{H}_+ = \vec{J}_s \tag{2.178}$$

$$\hat{n} \cdot \vec{B}_+ = 0 \tag{2.179}$$

A similar analysis can now be made to derive the boundary conditions for a perfect magnetic conductor:

$$-\hat{n} \times \vec{E}_+ = \vec{J}_m \tag{2.180}$$

$$\hat{n} \cdot \vec{D}_+ = 0 \tag{2.181}$$

$$\hat{n} \times \vec{H}_+ = 0 \tag{2.182}$$

$$\hat{n} \cdot \vec{B}_+ = q_m \tag{2.183}$$

where J_m is the magnetic surface current density and q_m is the surface magnetic charge density (usually zero). The perfect magnetic conductor is defined as a material inside of which time-varying electric and magnetic fields vanish when illuminated by an electromagnetic wave. In addition any tangential component of the magnetic field also disappears. Any (hypothetical) magnetic charges within the material quickly move to the surface where they reside as an infinitesimal layer forming a magnetic surface current density. Although such metamaterials do not physically exist, they are used here to develop the electromagnetic equivalents for a

range of real-world materials.

For completeness, the situation where the second medium is not perfectly conducting can now be considered. It needs to be recognised however that:

- There are now fields on both sides of the boundary
- Under normal conditions there are no surface charges on the boundary

These provide the standard boundary conditions for a boundary between finite conductivity media with no surface charges or currents, namely:

$$\hat{n} \times \left(\vec{E}_+ - \vec{E}_-\right) = 0 \tag{2.184}$$

$$\hat{n} \cdot \left(\vec{D}_+ - \vec{D}_-\right) = 0 \tag{2.185}$$

$$\hat{n} \times \left(\vec{H}_+ - \vec{H}_-\right) = \vec{J} \tag{2.186}$$

$$\hat{n} \cdot \left(\vec{B}_+ - \vec{B}_-\right) = 0 \tag{2.187}$$

2.2.3 Combination Sheet

In order to model realistic materials it is essential that both the electric and magnetic currents within the material are taken into account (as seen in equations (2.131)-(2.138)). Although realistic materials span a wide range of properties it is sometimes assumed that the material is a perfect electrical conductor. In such a case the material cannot support any magnetic currents (L=0) and so the equations are considerably simplified. In reality however any material has both magnetic and electrical properties and stimulation of currents in one property result in a current generation in the other. As a result the E and H fields become coupled, meaning that the strength of the E and H fields scattered by the induced currents are each effected by the other.

It is natural to consider a hypothetical realisation of a realistic material as being a combination of both electric and magnetic properties. This is called a combination sheet that comprises an infinitesimal electrically resistive sheet combined with an infinitesimal magnetically conductive sheet [87].

The resistivity of such a sheet is called its sheet-resistance Rs and extends the

discussion on resistivity earlier in the chapter (Figure 2.11). Its utility compared to resistivity is that it can be directly and conveniently measured experimentally using, for example, a four-point probe.

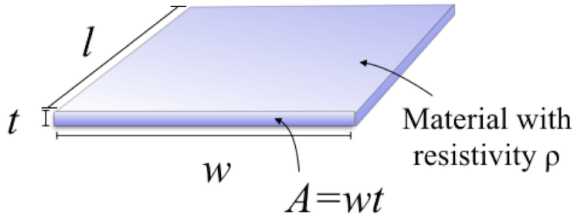


Figure 2.11: Sheet Resistance

The cross sectional area of a material can be defined in terms of its thickness and width. The resistivity can then be combined with the thickness of the sheet and so the resistance is given by:

$$R = \frac{\rho}{t}\frac{L}{W} = R_s\frac{L}{W} \tag{2.188}$$

where Rs is the sheet resistance. From this, if the sheet thickness is known, the material resistivity can be calculated by multiplying the sheet resistance by the film thickness and so

$$R_s = \frac{\rho}{t} = \frac{1}{\sigma t} \tag{2.189}$$

The units for sheet resistance are unusual and can best be understood by considering the units in (2.188). They are (*ohms.metre*^{2} / *metre*) / *metre* or ohms per square ($\Omega/\square$).

The boundary conditions for such a sheet arise from those in (2.184) to (2.187) and by considering the extreme case where t approaches zero. In a material that is not perfectly conducting, as t approaches zero, σ increases such that R_s is finite in the limit. For comparison, if $R_s = 0$ then any derivations represent the perfectly conducting case and when $R_s = \infty$ the sheet is perfectly transparent.

Consider the boundary conditions for the electrically resistive sheet within the combination sheet separately from the magnetically conductive counterpart. The

boundary conditions (from equations (2.184) and (2.186) are:

$$\begin{aligned} \hat{n} \times \vec{E}_+ - \hat{n} \times \vec{E}_- &= 0 \\ \hat{n} \times \vec{H}_+ - \hat{n} \times \vec{H}_- &= \vec{J} \end{aligned} \tag{2.190}$$

and as

$$\vec{E}_+^{\tan} = \vec{E}_-^{\tan} = \frac{\vec{J}_s}{\sigma} = R_s \vec{J}_s \tag{2.191}$$

which can be written in general as:

$$\hat{n} \times \left(\hat{n} \times \vec{E}_\pm \right) = -R_s \vec{J}_s \tag{2.192}$$

and so:

$$\hat{n} \times \left(\hat{n} \times \left[\vec{E}_+ + \vec{E}_- \right] \right) = -2R_s \vec{J} \tag{2.193}$$

The boundary conditions for the magnetically conductive sheet can also be written as:

$$\begin{aligned} \hat{n} \times \vec{H}_+ - \hat{n} \times \vec{H}_- &= 0 \\ \hat{n} \times \vec{E}_+ - \hat{n} \times \vec{E}_- &= -\vec{M} \end{aligned} \tag{2.194}$$

with

$$\hat{n} \times \left[\hat{n} \times \vec{H}_\pm \right] = -R_m \vec{M} \tag{2.195}$$

$$\hat{n} \times \left(\hat{n} \times \left[\vec{H}_+ + \vec{H}_- \right] \right) = -2R_m \vec{M} \tag{2.196}$$

The conditions can now be examined in terms of $\hat{x}$ and $\hat{y}$ and using the coordinate system in the plane of the scatterer. The total contributions to the fields from both components of the combination plate are therefore:

$$E_+^x + E_-^x = -2R_s \left(H_+^y - H_-^y \right) \tag{2.197}$$

$$E_+^x - E_-^x = -\frac{1}{2R_m} \left(H_+^y + H_-^y \right) \tag{2.198}$$

$$E_+^y + E_-^y = 2R_s \left(H_+^x - H_-^x \right) \tag{2.199}$$

$$E_+^y - E_-^y = \frac{1}{2R_m} \left(H_+^x + H_-^x \right) \tag{2.200}$$

and so combining the x and y components in pairs we have:

$$E_+^x = -R_s\left(1+\frac{1}{4R_sR_m}\right)H_+^y + R_s\left(1-\frac{1}{4R_sR_m}\right)H_-^y \tag{2.201}$$

$$E_-^x = -R_s\left(1-\frac{1}{4R_sR_m}\right)H_+^y + R_s\left(1+\frac{1}{4R_sR_m}\right)H_-^y \tag{2.202}$$

$$E_+^y = R_s\left(1+\frac{1}{4R_sR_m}\right)H_+^x - R_s\left(1-\frac{1}{4R_sR_m}\right)H_-^x \tag{2.203}$$

$$E_-^y = R_s\left(1-\frac{1}{4R_sR_m}\right)H_+^x - R_s\left(1+\frac{1}{4R_sR_m}\right)H_-^x \tag{2.204}$$

These equations describe the surface fields for the combination sheet and show how in general the scattered electric field is formed from a tight coupling between the material's electrical resistivity R_s and magnetic conductivity R_m. It can be seen from these equations that a special case occurs when:

$$4R_sR_m = 1 \tag{2.205}$$

In which case equations (2.201) through (2.204) become:

$$E_+^x = -2R_sH_+^y \tag{2.206}$$

$$E_-^x = 2R_sH_-^y \tag{2.207}$$

$$E_+^y = 2R_sH_+^x \tag{2.208}$$

$$E_-^y = -2R_sH_-^x \tag{2.209}$$

In this case the combination sheet is completely opaque (terms are either all + or all -) and is simply an impedance sheet with intrinsic surface impedance of $\eta = 2R_s$ on each side of the sheet. These are called the Leontovich Boundary Conditions [88].

As can be seen, the general solutions of the combination sheet are tightly coupled. However an exception occurs when the sheet is completely planar. Consider that the total electric field at the surface of the combination sheet is a sum of the incident field and also the electric fields from both the resistive and combination

sheets. In other words:

$$\vec{E} = \vec{E}^i + \vec{E}^m + \vec{E}^e \tag{2.210}$$

If one considers only the boundary conditions for the magnetically conductive sheet and assumes symmetry about the plane of the sheet, i.e. from (2.194):

$$\hat{n} \times \vec{E}^m_+ = \frac{\vec{M}}{2},\ \hat{n} \times \vec{E}^m_- = -\frac{\vec{M}}{2} \tag{2.211}$$

Then it follows that

$$\hat{n} \times \left\{ \hat{n} \times \left[\vec{E}^m_+ + \vec{E}^m_- \right] \right\} = 0 \tag{2.212}$$

From (2.193) the boundary conditions for an electrically resistive sheet can be combined with (2.210) to provide:

$$\hat{n} \times \left(\hat{n} \times \left[\left(\vec{E}^i + \vec{E}^m + \vec{E}^e \right)_+ + \left(\vec{E}^i + \vec{E}^m + \vec{E}^e \right)_- \right] \right) = -2R_s \vec{J} \tag{2.213}$$

and from (2.190) and (2.211):

$$\hat{n} \times \left(\hat{n} \times \left[\vec{E}^i_+ + \vec{E}^e_+ + \vec{E}^i_- + \vec{E}^e_- \right] \right) = -2R_s \hat{n} \times \left(\vec{H}^e_+ - \vec{H}^e_- \right) \tag{2.214}$$

This equation shows that for a planar configuration of the combination sheet the boundary conditions for the resistive sheet are completely independent to the conditions set by the conductive sheet. Performing the same analysis for the boundary conditions of the conductive sheet reveals that there are also no electrically resistive dependencies. In other words the two sheets are completely decoupled and can be treated in isolation.

2.2.4 Translation into Facet Coordinates

Before moving on to the derivation of the surface currents on a facet, it is worth spending a little time to simplify the coordinate systems used. As was shown in the previous section, induced surface currents exist only as a sheet along the surface of the facet. They therefore have no components in the facet normal direction. As a result the analysis, and also the software implementation, can be simplified

considerably by performing a translation of the global coordinate system into local facet coordinates. If only direction vectors are considered (Figure 2.12) then the

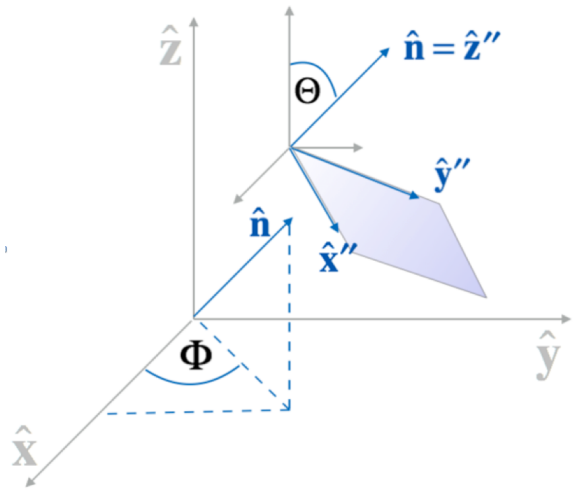


Figure 2.12: Coordinate Transformation From Global Coordinates to the Facet

coordinate conversion can be obtained simply through a rotation matrix. This is achieved by multiplying the vector by:

$$\vec{v''} = \vec{v}\begin{bmatrix} \cos\Phi & \sin\Phi & 0 \\ -\sin\Phi & \cos\Phi & 0 \\ 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} \cos\Theta & 0 & -\sin\Theta \\ 0 & 1 & 0 \\ \sin\Theta & 0 & \cos\Theta \end{bmatrix} \tag{2.215}$$

where $\vec{v}$ is the direction vector in global coordinates and $\vec{v''}$ is the direction vector in the local coordinate system of the facet defined by:

$$\Phi = \tan^{-1}\left(\frac{n_y}{n_x}\right), \Theta = \cos^{-1}\left(\hat{z}\cdot\hat{n}\right) \tag{2.216}$$

The required transformation from global to local coordinates is therefore:

$$\vec{v''} = \vec{v}\begin{bmatrix} \cos\Phi\cos\Theta & \sin\Phi & -\cos\Phi\sin\Theta \\ -\sin\Phi\cos\Theta & \cos\Phi & \sin\Phi\sin\Theta \\ 0 & 0 & \sin\Theta \end{bmatrix} \tag{2.217}$$

2.2.5 Transmission and Reflection Coefficients

The formulas for the surface currents are now developed for an EM ray incident on a reflecting sheet. Figure 2.13 presents the angles and coordinates used with subscripts i, s and t representing the incident ray, scattered ray and transmission ray respectively. θ is the angle of incidence of the incident ray and ϕ is an angle from the (arbitrarily defined) x-axis in the plane of the surface. The incident E-field can be defined in terms of its polarisation relative to the scattering surface:

$$\vec{E}^i = \left(\vec{E}^i_\theta \hat{\theta} + \vec{E}^i_\phi \hat{\phi}\right) e^{-j\vec{k}_i \cdot \vec{r}_i} \tag{2.218}$$

with $\hat{\theta}$ being the unit vector representing polarisations that are parallel to the plane

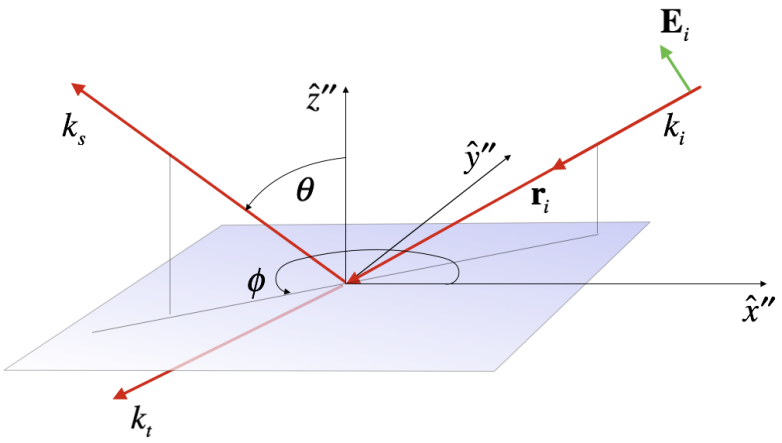


Figure 2.13: Coordinate System for a Ray Incident on a Reflecting Sheet

of incidence[2] and $\hat{\phi}$ being the unit vector representing polarisations that are perpendicular to the plane of incidence (Figure 2.14). As the parallel and perpendicular vectors are orthogonal it is possible to consider individual polarisations in isolation and then recombine them afterwards. This is useful as the development can be significantly simplified by considering only a single polarisation.

Before proceeding further it is worth summarising the boundary conditions for the combination sheet. The conditions relating to the electrically resistive surface

[2]The plane of incidence is the plane that contains the incident, scattered and transmitted rays.

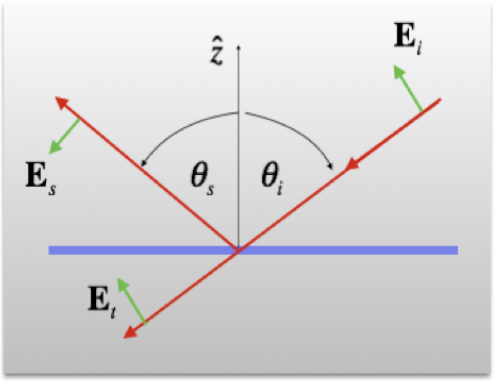


Figure 2.14: Polarisation is Defined Relative to the Plane of Incidence

are ((2.190), (2.192) and (2.193):

$$\begin{aligned} \hat{n}\times\left(\vec{E}_+ - \vec{E}_-\right) &= 0 \\ \hat{n}\times\left(\hat{n}\times\vec{E}_\pm\right) &= -R_s\vec{J}_s \\ \hat{n}\times\left(\vec{H}_+ - \vec{H}_-\right) &= \vec{J}_s \end{aligned} \tag{2.219}$$

and the boundary conditions for the magnetically conductive surface are ((2.194), (2.195) and (2.196)):

$$\begin{aligned} \hat{n}\times\left(\vec{H}_+ - \vec{H}_-\right) &= 0 \\ \hat{n}\times\left(\hat{n}\times\vec{H}_\pm\right) &= -R_m\vec{M} \\ \hat{n}\times\left(\vec{E}_+ - \vec{E}_-\right) &= -\vec{M} \end{aligned} \tag{2.220}$$

Consider the conditions at the point when the incident ray hits the surface. At this location $z=0$. From Figure 2.14 the total E field intensity on the positive side are composed of the incident E and the scattered E field. Equally the total E field intensity on the negative side comprise only the transmission E field. In other words at the boundary:

$$\begin{aligned} \vec{E}_+ &= \left(\vec{E}^i + \vec{E}^s\right)\Big|_{z=0} \\ \vec{E}_- &= \vec{E}^t\Big|_{z=0} \end{aligned} \tag{2.221}$$

Equally, the magnetic field intensity on the positive side must be equal to the field

intensity on the negative side:

$$\begin{aligned} \vec{H}_+ &= \left(\vec{H}^i + \vec{H}^s\right)\Big|_{z=0} \\ \vec{H}_- &= \vec{H}^t\Big|_{z=0} \end{aligned} \tag{2.222}$$

2.2.5.1 Reflection and Transmission Coefficients - Parallel Polarisation

Consider the case where the electric field of an incident ray is parallel to the plane of incidence, (Figure 2.14). Starting with the definition of the incident E field from (2.218):

$$\vec{E}^i = \left(\vec{E}^i_\theta \hat{\theta} + \vec{E}^i_\phi \hat{\phi}\right) e^{-j\vec{k}_i \cdot \vec{r}_i} \tag{2.223}$$

In this situation there is no component of the electric field that is perpendicular and therefore:

$$\vec{E}^i_\phi = 0 \tag{2.224}$$

which means that equation (2.223) becomes:

$$\vec{E}^i = \vec{E}^i_\theta \hat{\theta} e^{-j\vec{k}_i \cdot \vec{r}_i} \tag{2.225}$$

From Figure 2.14 and remembering that only the coordinates in the plane of incidence are being considered (i.e. no y coordinate), the incident E field can be defined in terms of its magnitude E_0 as:

$$\vec{E}^i = E_0 \left[-\hat{x}'' \cos\theta_i + \hat{z}'' \sin\theta_i\right] e^{-jk\left[-x'' \sin\theta_i - z'' \cos\theta_i\right]} \tag{2.226}$$

with $k = 2\pi/\lambda$ as always and with the double prime superscript denoting that the coordinates are in the local facet plane. The subscripts θ_i and ϕ_i denote incidence angles and so are always in the plane of the facet. Also:

$$\vec{H}^i = \hat{y}'' \frac{E_0}{Z_0} e^{-jk\left[-x'' \sin\theta_i - z'' \cos\theta_i\right]} \tag{2.227}$$

with Z_0 being the wave impedance in free space. Setting two scaling functions for the parallel polarisation transmission coefficient and reflection coefficient as $T_{\parallel}$ and $\Gamma_{\parallel}$ respectively provides:

$$\vec{E}^t = T_{\parallel} E_0 \left[-\hat{x}'' \cos\theta_i + \hat{z}'' \sin\theta_i\right] e^{-jk\left[-x'' \sin\theta_i - z'' \cos\theta_i\right]} \tag{2.228}$$

$$\vec{H}^t = \hat{y}'' T_{\parallel} \frac{E_0}{Z_0} e^{-jk\left[-x'' \sin\theta_i - z'' \cos\theta_i\right]} \tag{2.229}$$

and

$$\vec{E}^s = \Gamma_{\parallel} E_0 \left[-\hat{x}'' \cos\theta_i - \hat{z}'' \sin\theta_i\right] e^{-jk\left[-x'' \sin\theta_i + z'' \cos\theta_i\right]} \tag{2.230}$$

$$\vec{H}^s = -\hat{y}'' \Gamma_{\parallel} \frac{E_0}{Z_0} e^{-jk\left[-x'' \sin\theta_i + z'' \cos\theta_i\right]} \tag{2.231}$$

which yields the E and H fields at the exact location of the boundary $z=0$. Using the boundary conditions for the electrically resistive sheet from (2.219) combined with (2.221) provides:

$$\hat{n}'' \times \vec{E}_+ - \hat{n}'' \times \vec{E}_- = 0 \tag{2.232}$$

$$\hat{n}'' \times \left(\vec{E}^i + \vec{E}^s\right)\Big|_{z=0} = \hat{n}'' \times \vec{E}^t\Big|_{z=0} \tag{2.233}$$

and therefore substituting in the terms derived in (2.226), (2.228) and (2.230) produces:

$$\begin{aligned} \hat{n}'' \times & \left(\begin{array}{c} \left(E_0 \left[-\hat{x}'' \cos\theta_i + \hat{z}'' \sin\theta_i\right] e^{-jk\left[-x'' \sin\theta_i\right]}\right) \\ + \left(\Gamma_{\parallel}^e E_0 \left[-\hat{x}'' \cos\theta_i - \hat{z}'' \sin\theta_i\right] e^{-jk\left[-x'' \sin\theta_i\right]}\right) \end{array} \right) \\ & = \hat{n}'' \times \left(T_{\parallel}^e E_0 \left[-\hat{x}'' \cos\theta_i + \hat{z}'' \sin\theta_i\right] e^{-jk\left[-x'' \sin\theta_i\right]}\right)\Big|_{z=0} \end{aligned} \tag{2.234}$$

where the e superscript denotes that the coefficient is related to the electrically resistive sheet. The above can be further simplified by recognising that:

$$\hat{n}'' \times \hat{z}'' = 0 \qquad \hat{n}'' \times \hat{x}'' = \hat{y}'' \tag{2.235}$$

can be substituted into

$$\begin{aligned} -\hat{y}'' \cos\theta_i E_0 e^{-jk\left[-x''\sin\theta_i\right]} - \hat{y}'' \cos\theta_i \Gamma_{\parallel}^{e} E_0 e^{-jk\left[-x''\sin\theta_i\right]} \\ = -\hat{y}'' \cos\theta_i T_{\parallel}^{e} E_0 e^{-jk\left[-x''\sin\theta_i\right]} \end{aligned} \tag{2.236}$$

which conveniently provides the relationship between the parallel reflection and transmission coefficients for an electrically resistive sheet:

$$1 + \Gamma_{\parallel}^{e} = T_{\parallel}^{e} \tag{2.237}$$

Using the boundary conditions for the magnetically conductive sheet (2.220) combined with (2.221) provides:

$$\hat{n} \times \vec{H}_{+} - \hat{n} \times \vec{H}_{-} = 0 \tag{2.238}$$

$$\hat{n} \times \left(\vec{H}^{i} + \vec{H}^{s}\right)\Big|_{z=0} = \hat{n} \times \vec{H}^{t}\Big|_{z=0} \tag{2.239}$$

and so from (2.227), (2.229) and (2.231):

$$\begin{aligned} \hat{n}'' \times \Bigg(& \left(\hat{y}'' \frac{E_0}{Z_0} e^{-jk\left[-x''\sin\theta_i - z''\cos\theta_i\right]}\right) \\ & + \left(-\hat{y}'' \Gamma_{\parallel}^{m} \frac{E_0}{Z_0} e^{-jk\left[-x''\sin\theta_i + z''\cos\theta_i\right]}\right) \Bigg)\Bigg|_{z=0} \\ = \hat{n}'' \times & \left(\hat{y}'' T_{\parallel}^{m} \frac{E_0}{Z_0} e^{-jk\left[-x''\sin\theta_i - z''\cos\theta_i\right]}\right)\Bigg|_{z=0} \end{aligned} \tag{2.240}$$

$$\left(-\hat{x}'' \frac{E_0}{Z_0} e^{-jk\left[-x''\sin\theta_i\right]}\right) + \left(\hat{x}'' \Gamma_{\parallel}^{m} \frac{E_0}{Z_0} e^{-jk\left[-x''\sin\theta_i\right]}\right) = -\hat{x}'' T_{\parallel}^{m} \frac{E_0}{Z_0} e^{-jk\left[-x''\sin\theta_i\right]} \tag{2.241}$$

which provides the relationship between the parallel reflection and transmission coefficients for the magnetically conductive sheet:

$$1 - \Gamma_{\parallel}^{m} = T_{\parallel}^{m} \tag{2.242}$$

2.2.5.2 Reflection and Transmission Coefficients - Perpendicular Polarisation

The reflection and transmission coefficients for the perpendicular component of the E field will now be derived. The analysis is very similar to the previous section but is here for completeness. As before the analysis starts off with the definition of the incident E-field from (2.218):

$$\vec{E}^i = \left(\vec{E}^i_\theta \hat{\theta} + \vec{E}^i_\phi \hat{\phi}\right) e^{-j\vec{k}_i \cdot \vec{r}_i} \tag{2.243}$$

This time however the perpendicular component of the E field is:

$$\vec{E}^i = \left(\vec{E}^i_\phi \hat{\phi}\right) e^{-j\vec{k}_i \cdot \vec{r}_i} \tag{2.244}$$

and from Figure 2.13 it follows that the incident fields can be given by:

$$\vec{E}^i = -\hat{y}'' E_0 e^{-jk\left[-x''\sin\theta - z''\cos\theta\right]} \tag{2.245}$$

and

$$\vec{H}^i = \frac{E_0}{Z_0}\left[-\hat{x}''\cos\theta_i + \hat{z}''\sin\theta_i\right] e^{-jk\left[-x''\sin\theta_i - z''\cos\theta_i\right]} \tag{2.246}$$

Establishing the perpendicular field reflection coefficient as $\Gamma_\perp$ and the perpendicular field transmission coefficient as $T_\perp$ then fields for $\vec{E}^t$, $\vec{H}^t$, and $\vec{E}^s$ can be written as:

$$\vec{E}^t = -\hat{y}'' T_\perp E_0 e^{-jk\left[-x''\sin\theta_i - z''\cos\theta_i\right]} \tag{2.247}$$

$$\vec{H}^t = T_\perp \frac{E_0}{Z_0}\left[-\hat{x}''\cos\theta_i + \hat{z}''\sin\theta_i\right] e^{-jk\left[-x''\sin\theta_i - z''\cos\theta_i\right]} \tag{2.248}$$

$$\vec{E}^s = -\hat{y}'' \Gamma_\perp E_0 e^{-jk\left[-x''\sin\theta_i + z''\cos\theta_i\right]} \tag{2.249}$$

$$\vec{H}^s = \Gamma_\perp \frac{E_0}{Z_0}\left[\hat{x}''\cos\theta_i + \hat{z}''\sin\theta_i\right] e^{-jk\left[-x''\sin\theta_i + z''\cos\theta_i\right]} \tag{2.250}$$

Using the boundary conditions for the electrically resistive sheet from (2.219) combined with (2.221) we get:

$$\hat{n} \times \vec{E}_+ - \hat{n} \times \vec{E}_- = 0 \tag{2.251}$$

$$\hat{n} \times \left(\left(\vec{E}^i + \vec{E}^s \right)\Big|_{z=0} \right) = \hat{n} \times \vec{E}^t\Big|_{z=0} \tag{2.252}$$

Substituting in the geometrical terms for E from (2.245), (2.247) and (2.249) gives:

$$\begin{aligned} \hat{n}'' \times \Bigg(\Bigg(& \left(-\hat{y}'' E_0 e^{-jk\left[-x'' \sin\theta - z'' \cos\theta\right]} \right) \\ & + \left(-\hat{y}'' \Gamma_\perp^e E_0 e^{-jk\left[-x'' \sin\theta_i + \hat{z}'' \cos\theta_i\right]} \right) \Bigg) \Bigg|_{z=0} \Bigg) \\ = \hat{n}'' \times & \left(-\hat{y}'' T_\perp^e E_0 e^{-jk\left[-x'' \sin\theta_i - z'' \cos\theta_i\right]} \right) \Big|_{z=0} \end{aligned} \tag{2.253}$$

$$\begin{aligned} & \hat{n}'' \times \left(\left(-\hat{y}'' E_0 e^{-jk\left[-x'' \sin\theta\right]} \right) + \left(-\hat{y}'' \Gamma_\perp^e E_0 e^{-jk\left[-x'' \sin\theta_i\right]} \right) \right) \\ & = \hat{n}'' \times \left(-\hat{y}'' T_\perp^e E_0 e^{-jk\left[-x'' \sin\theta_i\right]} \right) \end{aligned} \tag{2.254}$$

where the e superscript denotes that the coefficient is related to the electrically resistive sheet. The perpendicular reflection and transmission coefficients of the electrically resistive sheet are therefore related by:

$$1 + \Gamma_\perp^e = T_\perp^e \tag{2.255}$$

The boundary conditions for the magnetically conductive sheet (2.220) with (2.221) can now be used to get:

$$\hat{n} \times \vec{H}_+ - \hat{n} \times \vec{H}_- = 0 \tag{2.256}$$

$$\hat{n} \times \left(\vec{H}^i + \vec{H}^s \right)\Big|_{z=0} = \hat{n} \times \vec{H}^t\Big|_{z=0} \tag{2.257}$$

and so using (2.246), (2.248) and (2.250):

$$\begin{aligned} & \left(\hat{n}'' \times -\hat{x}'' \right) \cos\theta_i \frac{E_0}{Z_0} e^{-jk\left[-x'' \sin\theta_i\right]} \\ & \quad + \left(\hat{n}'' \times \hat{x}'' \right) \cos\theta_i \Gamma_\perp^m \frac{E_0}{Z_0} e^{-jk\left[-x'' \sin\theta_i\right]} \\ & \quad - \left(\hat{n}'' \times -\hat{x}'' \right) \cos\theta_i T_\perp^m \frac{E_0}{Z_0} e^{-jk\left[-x'' \sin\theta_i\right]} = 0 \end{aligned} \tag{2.258}$$

The perpendicular reflection and transmission coefficients of the magnetically conductive sheet are therefore related by:

$$1 - \Gamma_{\perp}^{m} = T_{\perp}^{m} \tag{2.259}$$

2.2.5.3 Parallel Polarisation Coefficients for the Electric Field

It has been shown that the magnetic field boundary condition for the resistive sheet is (2.219):

$$\hat{n} \times \vec{H}_{+} - \hat{n} \times \vec{H}_{-} = \vec{J}_s \tag{2.260}$$

$$\vec{H}_{+} = \left(\vec{H}^{i} + \vec{H}^{s}\right)\Big|_{z=0} \tag{2.261}$$

$$\vec{H}_{-} = \vec{H}^{t}\Big|_{z=0} \tag{2.262}$$

and therefore

$$\hat{n} \times \left(\left(\vec{H}^{i} + \vec{H}^{s}\right)\Big|_{z=0}\right) - \hat{n} \times \left(\vec{H}^{t}\Big|_{z=0}\right) = \vec{J}_s \tag{2.263}$$

Substituting for the geometrical values for H from (2.222) and (2.227), (2.229) and (2.231) to provide:

$$\begin{aligned} \hat{n}'' \times \left(\hat{y}'' \frac{E_0}{Z_0} e^{-jk\left[-x'' \sin\theta_i\right]} - \hat{y}'' \Gamma_{\parallel}^{e} \frac{E_0}{Z_0} e^{-jk\left[-x'' \sin\theta_i\right]}\right) \\ -\hat{n}'' \times \left(\hat{y}'' T_{\parallel}^{e} \frac{E_0}{Z_0} e^{-jk\left[-x'' \sin\theta_i\right]}\right) = \vec{J}_s \end{aligned} \tag{2.264}$$

$$\begin{aligned} \hat{n}'' \times \hat{y}'' \frac{E_0}{Z_0} \left(e^{-jk\left[-x'' \sin\theta_i\right]} - \Gamma_{\parallel}^{e} e^{-jk\left[-x'' \sin\theta_i\right]}\right) \\ -\hat{n}'' \times \hat{y}'' \frac{E_0}{Z_0} \left(T_{\parallel}^{e} e^{-jk\left[-x'' \sin\theta_i\right]}\right) = \vec{J}_s \end{aligned} \tag{2.265}$$

$$\hat{x}'' \frac{E_0}{Z_0} e^{jkx'' \sin\theta_i} \left(1 - \Gamma_{\parallel}^{e} - T_{\parallel}^{e}\right) = \vec{J}_s \tag{2.266}$$

and from equation (2.219)

$$\hat{n}'' \times \left(\hat{n}'' \times \vec{E}_{\pm}\right) = -R_s \hat{x}'' e^{jkx'' \sin\theta_i} \frac{E_0}{Z_0} \left(1 - \Gamma_{\parallel}^{e} - T_{\parallel}^{e}\right) \tag{2.267}$$

Using equations (2.221) with (2.228) this can be rewritten as:

$$\begin{aligned}\hat{n}'' \times \left(\hat{n}'' \times \left(-T_{\parallel}^{e} E_0 \left[-\hat{x}'' \cos\theta_i + \hat{z}'' \sin\theta_i\right] e^{-jk\left[-x'' \sin\theta_i\right]}\right)\right) \\ = -\hat{x}'' E_0 e^{jkx'' \sin\theta_i} \frac{R_s}{Z_0} \left(1 - \Gamma_{\parallel}^{e} - T_{\parallel}^{e}\right)\end{aligned} \tag{2.268}$$

$$\begin{aligned}\hat{n}'' \times \left(\hat{n}'' \times \left(T_{\parallel}^{e} E_0 \hat{x}'' \cos\theta_i e^{-jk\left[-x'' \sin\theta_i\right]} + T_{\parallel}^{e} E_0 \hat{z}'' \sin\theta_i e^{-jk\left[-x'' \sin\theta_i\right]}\right)\right) \\ = -\hat{x}'' E_0 e^{jkx'' \sin\theta_i} \frac{R_s}{Z_0} \left(1 - \Gamma_{\parallel}^{e} - T_{\parallel}^{e}\right)\end{aligned} \tag{2.269}$$

$$-T_{\parallel}^{e} E_0 \hat{x}'' \cos\theta_i e^{-jk\left[-x'' \sin\theta_i\right]} = -\hat{x}'' E_0 e^{jkx'' \sin\theta_i} \frac{R_s}{Z_0} \left(1 - \Gamma_{\parallel}^{e} - T_{\parallel}^{e}\right) \tag{2.270}$$

$$T_{\parallel}^{e} \cos\theta_i = \frac{R_s}{Z_0} \left(1 - \Gamma_{\parallel}^{e} - T_{\parallel}^{e}\right) \tag{2.271}$$

This equation when combined with (2.237) produces expressions for the parallel components of the transmission and reflection coefficients of the E-field, which will be used later on:

$$T_{\parallel}^{e} = \frac{2R_s}{Z_0 \cos\theta_i + 2R_s} \tag{2.272}$$

$$\Gamma_{\parallel}^{e} = \frac{-Z_0 \cos\theta_i}{Z_0 \cos\theta_i + 2R_s} \tag{2.273}$$

2.2.5.4 Perpendicular Polarisation Coefficients for the Electric Field

From (2.219) and (2.222) :

$$\hat{n} \times \vec{H}_+ - \hat{n} \times \vec{H}_- = \vec{J}_s \tag{2.274}$$

$$\vec{H}_+ = \left(\vec{H}^i + \vec{H}^s\right)\Big|_{z=0} \tag{2.275}$$

$$\vec{H}_- = \vec{H}^t\Big|_{z=0} \tag{2.276}$$

and therefore

$$\hat{n} \times \left(\left(\vec{H}^i + \vec{H}^s\right)\Big|_{z=0}\right) - \hat{n} \times \left(\vec{H}^t\Big|_{z=0}\right) = \vec{J}_s \tag{2.277}$$

Using the geometrical derivations from (2.246), (2.248) and (2.250):

$$\hat{n}'' \times \left(\begin{array}{c} \left(\frac{E_0}{Z_0} \left[-\hat{x}'' \cos\theta_i + \hat{z}'' \sin\theta_i \right] e^{-jk\left[-x'' \sin\theta_i \right]} \right) \\ + \left(\Gamma_\perp^e \frac{E_0}{Z_0} \left[\hat{x}'' \cos\theta_i + \hat{z}'' \sin\theta_i \right] e^{-jk\left[-x'' \sin\theta_i \right]} \right) \end{array} \right) \\ -\hat{n}'' \times \left(T_\perp^e \frac{E_0}{Z_0} \left[-\hat{x}'' \cos\theta_i + \hat{z}'' \sin\theta_i \right] e^{-jk\left[-x'' \sin\theta_i \right]} \right) = \vec{J}_s \tag{2.278}$$

$$\hat{n}'' \times \left(\begin{array}{c} \left(\frac{E_0}{Z_0} \left[-\hat{x}'' \cos\theta_i + \hat{z}'' \sin\theta_i \right] e^{-jk\left[-x'' \sin\theta_i \right]} \right) \\ + \left(\Gamma_\perp^e \frac{E_0}{Z_0} \left[\hat{x}'' \cos\theta_i + \hat{z}'' \sin\theta_i \right] e^{-jk\left[-x \sin\theta_i \right]} \right) \end{array} \right) \\ -\hat{n}'' \times \left(T_\perp^e \frac{E_0}{Z_0} \left[-\hat{x}'' \cos\theta_i + \hat{z}'' \sin\theta_i \right] e^{-jk\left[-x'' \sin\theta_i \right]} \right) = \vec{J}_s \tag{2.279}$$

and so as $\hat{n}'' \times \hat{z}'' = 0$:

$$\hat{n}'' \times \left(-\hat{x}'' \frac{E_0}{Z_0} \cos\theta_i e^{-jk\left[-x'' \sin\theta_i \right]} + \hat{x}'' \Gamma_\perp^e \frac{E_0}{Z_0} \cos\theta_i e^{-jk\left[-x'' \sin\theta_i \right]} \right) \\ -\hat{n}'' \times \left(-\hat{x}'' T_\perp^e \frac{E_0}{Z_0} \cos\theta_i e^{-jk\left[-x'' \sin\theta_i \right]} \right) = \vec{J}_s \tag{2.280}$$

$$-\hat{y}'' \frac{E_0}{Z_0} \cos\theta_i e^{-jk\left[-x'' \sin\theta_i \right]} + \hat{y}'' \Gamma_\perp^e \frac{E_0}{Z_0} \cos\theta_i e^{-jk\left[-x'' \sin\theta_i \right]} \\ + \hat{y}'' T_\perp^e \frac{E_0}{Z_0} \cos\theta_i e^{-jk\left[-x'' \sin\theta_i \right]} = \vec{J}_s \tag{2.281}$$

$$\hat{y}'' \frac{E_0}{Z_0} \cos\theta_i \left(\Gamma_\perp^e + T_\perp^e - 1 \right) e^{-jk\left[-x'' \sin\theta_i \right]} = \vec{J}_s \tag{2.282}$$

This equation can now be plugged into the boundary condition (2.219)

$$\hat{n}'' \times \left(\hat{n}'' \times \overline{E}\left(\pm \right) \right) = -R_s \left(\hat{y}'' \frac{E_0}{Z_0} \cos\theta_i \left(\Gamma_\perp + T_\perp - 1 \right) e^{-jk\left[-x'' \sin\theta_i \right]} \right) \tag{2.283}$$

and from (2.221) and (2.247):

$$\hat{n}'' \times \left(\hat{n}'' \times \left(-\hat{y}'' T_\perp^e E_0 e^{-jk\left[-x'' \sin\theta_i \right]} \right) \right) = \\ -R_s \left(\hat{y}'' \frac{E_0}{Z_0} \cos\theta_i \left(\Gamma_\perp^e + T_\perp^e - 1 \right) e^{-jk\left[-x'' \sin\theta_i \right]} \right) \tag{2.284}$$

$$\hat{n}'' \times \left(\hat{n}'' \times \left(-\hat{y}'' T_\perp^e \right) \right) = -\hat{y}'' \frac{R_s}{Z_0} \cos\theta_i \left(\Gamma_\perp^e + T_\perp^e - 1 \right) \tag{2.285}$$

$$T_{\perp}^{e} = \frac{R_s}{Z_0}\cos\theta_i\left(1-\Gamma_{\perp}^{e}-T_{\perp}^{e}\right) \tag{2.286}$$

and using equation (2.255)

$$T_{\perp}^{e} = \frac{R_s}{Z_0}\cos\theta_i\left(2-2T_{\perp}^{e}\right) \tag{2.287}$$

which provides useful equations for the transmission and reflection coefficients for the perpendicular components of the E-fields:

$$T_{\perp}^{e} = \frac{2R_s\cos\theta_i}{2R_s\cos\theta_i + Z_0} \tag{2.288}$$

$$\Gamma_{\perp}^{e} = \frac{-Z_0}{2R_s\cos\theta_i + Z_0} \tag{2.289}$$

All the polarisation states for the E field have been derived and it is now possible to combine these together to determine the values for the electric surface currents on a scattering surface.

2.2.5.5 Parallel Polarisation Coefficients for the Magnetic Field

Equations (2.172) and (2.273) provided the solutions for the parallel transmission and reflection coefficients for the electric field of a wave incident of a surface with electrical resistivity R_s. Although the concept of magnetic conductivity is a mathematical construct it is a useful parameter to capture the properties of an illuminated surface. The construction follows the same process as outlined previously for the E-field. The fundamental difference however is that now the surface is a magnetically conductive plate rather than and electrically resistive plate. Recall that the parallel reflection and transmission coefficients for such a sheet are (2.242):

$$1-\Gamma_{\parallel}^{m} = T_{\parallel}^{m} \tag{2.290}$$

From the equation for the magnetic surface current density (2.220):

$$\hat{n}\times\left(\vec{E}_{+}-\vec{E}_{-}\right) = -\vec{J}_m \tag{2.291}$$

then by substituting from (2.221):

$$\hat{n}\times\left(\left(\vec{E}^{i}+\vec{E}^{s}\right)\Big|_{z=0}\right)-\hat{n}\times\left(\vec{E}^{t}\Big|_{z=0}\right)=-\vec{J}_{m} \tag{2.292}$$

and using (2.226), (2.228) and (2.230):

$$\begin{aligned}&\hat{n}''\times\left(\left(\begin{array}{l}\left(E_0\left[-\hat{x}''\cos\theta_i+\hat{z}''\sin\theta_i\right]e^{-jk\left[-x''\sin\theta_i-z''\cos\theta_i\right]}\right)\\+\left(\Gamma_{\parallel}^{m}E_0\left[-\hat{x}''\cos\theta_i-\hat{z}''\sin\theta_i\right]e^{-jk\left[-x''\sin\theta_i+z''\cos\theta_i\right]}\right)\end{array}\right)\Bigg|_{z=0}\right)\\&-\hat{n}''\times\left(\left(T_{\parallel}^{m}E_0\left[-\hat{x}''\cos\theta_i+\hat{z}''\sin\theta_i\right]e^{-jk\left[-x''\sin\theta_i-z''\cos\theta_i\right]}\right)\Big|_{z=0}\right)=-\vec{J}_{m}\end{aligned} \tag{2.293}$$

This equation can be solved at the boundary where z=0 and remembering that to give:

$$\hat{y}''E_0\cos\theta_i e^{-jk\left[-x''\sin\theta_i\right]}\left(+T_{\parallel}^{m}-\Gamma_{\parallel}^{m}-1\right)=-\vec{J}_{m} \tag{2.294}$$

From (2.220)

$$\hat{n}''\times\left(\hat{n}''\times\vec{H}_{\pm}\right)=R_m\left(\hat{y}''E_0\cos\theta_i e^{-jk\left[-x''\sin\theta_i\right]}\left(+T_{\parallel}^{m}-\Gamma_{\parallel}^{m}-1\right)\right) \tag{2.295}$$

From (2.222) and (2.229):

$$\vec{H}_{-}=\vec{H}^{t}\Big|_{z=0}=\hat{y}''T_{\parallel}\frac{E_0}{Z_0}e^{-jk\left[-x''\sin\theta_i-z''\cos\theta_i\right]}\Big|_{z=0} \tag{2.296}$$

$$\begin{aligned}&\hat{n}''\times\left(\hat{n}''\times\left(\hat{y}''T_{\parallel}^{m}\frac{E_0}{Z_0}e^{-jk\left[-x''\sin\theta_i-z''\cos\theta_i\right]}\right)\Big|_{z=0}\right)\\&=R_m\left(\hat{y}''E_0\cos\theta_i e^{-jk\left[-x''\sin\theta_i\right]}\left(T_{\parallel}^{m}-\Gamma_{\parallel}^{m}-1\right)\right)\end{aligned} \tag{2.297}$$

$$T_{\parallel}^{m}=-Z_0R_m\cos\theta_i\left(T_{\parallel}^{m}-\Gamma_{\parallel}^{m}-1\right) \tag{2.298}$$

and so using (2.290) expressions for the transmission and reflection parallel magnetic current coefficients in terms of the magnetic conductivity can be provided:

$$\Gamma_{\parallel}^{m}=\frac{1}{1-2Z_0R_m\cos\theta_i} \tag{2.299}$$

and

$$T_{\parallel}^{m} = \frac{2Z_0 R_m \cos\theta_i}{1 + 2Z_0 R_m \cos\theta_i} \tag{2.300}$$

2.2.5.6 Perpendicular Polarisation Coefficients for the Magnetic Field

Recall that the perpendicular reflection and transmission coefficients for a magnetically conductive sheet are (2.259)

$$1 - \Gamma_{\perp}^{m} = T_{\perp}^{m} \tag{2.301}$$

From the equation for the magnetic surface current density (2.220):

$$\hat{n} \times \left(\vec{E}_{+} - \vec{E}_{-}\right) = -\vec{J}_m \tag{2.302}$$

Then by substituting from (2.221) :

$$\hat{n} \times \left(\left(\vec{E}^{i} + \vec{E}^{s}\right)\Big|_{z=0}\right) - \hat{n} \times \left(\vec{E}^{t}\Big|_{z=0}\right) = -\vec{J}_m \tag{2.303}$$

The terms relating $\vec{E}^{i}$, $\vec{E}^{s}$ and $\vec{E}^{t}$ to the perpendicular H field for the magnetically conductive sheet can now be used (2.245), (2.247) and (2.249):

$$\begin{aligned}&\hat{n}'' \times \left(\left(\left(-\hat{y}'' E_0 e^{-jk\left[-x''\sin\theta - z''\cos\theta\right]}\right) + \left(-\hat{y}'' \Gamma_{\perp}^{m} E_0 e^{-jk\left[-x''\sin\theta_i + z''\cos\theta_i\right]}\right)\right)\Big|_{z=0}\right) \\ &-\hat{n}'' \times \left(-\hat{y}'' T_{\perp}^{m} E_0 e^{-jk\left[-x''\sin\theta_i - z''\cos\theta_i\right]}\Big|_{z=0}\right) = -\vec{J}_m\end{aligned} \tag{2.304}$$

At z=0 and as

$$\begin{aligned}&\left(\hat{n}'' \times -\hat{y}''\right) E_0 e^{-jk\left[-x''\sin\theta\right]} + \left(\hat{n}'' \times -\hat{y}''\right) \Gamma_{\perp}^{m} E_0 e^{-jk\left[-x''\sin\theta_i\right]} \\ &-\left(\hat{n}'' \times -\hat{y}''\right) T_{\perp}^{m} E_0 e^{-jk\left[-x''\sin\theta_i\right]} = -\vec{J}_m\end{aligned} \tag{2.305}$$

$$\hat{x}'' E_0 e^{-jk\left[-x''\sin\theta\right]} \left(1 + \Gamma_{\perp}^{m} - T_{\perp}^{m}\right) = -\vec{J}_m \tag{2.306}$$

From (2.220):

$$\hat{n} \times \left(\hat{n} \times \vec{H}_{\pm}\right) = -R_m \vec{J}_m \tag{2.307}$$

$$\hat{n}'' \times \left(\hat{n}'' \times \vec{H}_{\pm}\right) = R_m \left(\hat{x}'' E_0 e^{-jk\left[-x'' \sin\theta\right]} \left(1 + \Gamma_{\perp}^{m} - T_{\perp}^{m}\right)\right) \tag{2.308}$$

and using $\vec{H}^t$ as $\vec{H}_{\pm}$:

$$\hat{n}'' \times \left(\hat{n}'' \times \vec{H}^t\big|_{z=0}\right) = R_m \left(\hat{x}'' E_0 e^{-jk\left[-x'' \sin\theta\right]} \left(1 + \Gamma_{\perp}^{m} - T_{\perp}^{m}\right)\right) \tag{2.309}$$

Substituting for $\vec{H}^t$ from (2.248):

$$\begin{aligned} &\hat{n}'' \times \left(\hat{n}'' \times T_{\perp}^{m} \frac{E_0}{Z_0} \left[-\hat{x}'' \cos\theta_i + \hat{z}'' \sin\theta_i\right] e^{-jk\left[-x'' \sin\theta_i\right]}\right) \\ &= R_m \left(\hat{x}'' E_0 e^{-jk\left[-x'' \sin\theta\right]} \left(1 + \Gamma_{\perp}^{m} - T_{\perp}^{m}\right)\right) \end{aligned} \tag{2.310}$$

and so

$$T_{\perp}^{m} = \frac{Z_0 R_m \left(1 + \Gamma_{\perp}^{m} - T_{\perp}^{m}\right)}{\cos\theta_i} \tag{2.311}$$

which when combined with (2.301) provides expressions for the perpendicular reflections and transmission coefficients for an EM wave incident on a magnetically conductive sheet:

$$\Gamma_{\perp}^{m} = \frac{\cos\theta_i}{2Z_0 R_m + \cos\theta_i} \tag{2.312}$$

$$T_{\perp}^{m} = \frac{2Z_0 R_m}{\cos\theta_i + 2Z_0 R_m} \tag{2.313}$$

2.3 The Induced Surface Currents

So far this chapter has provided equations that determine the nature of the reflection and transmission of EM waves in terms of their polarisation and the material properties of the boundary surface. This analysis has only focussed on the interaction of the incident fields at the boundary to the surface. The next task is to use these formulations to derive equations that describe the induced surface currents on the boundary from an incident wave. The incident wave however will originate at a location away from the boundary surface and will therefore comprise phase information caused by its propagation from its source to the surface. EM modelling often requires that multiple surfaces be considered, each illuminated by the same originating source.

Consider Figure 2.15 which shows an incident EM ray hitting a triangular facet.

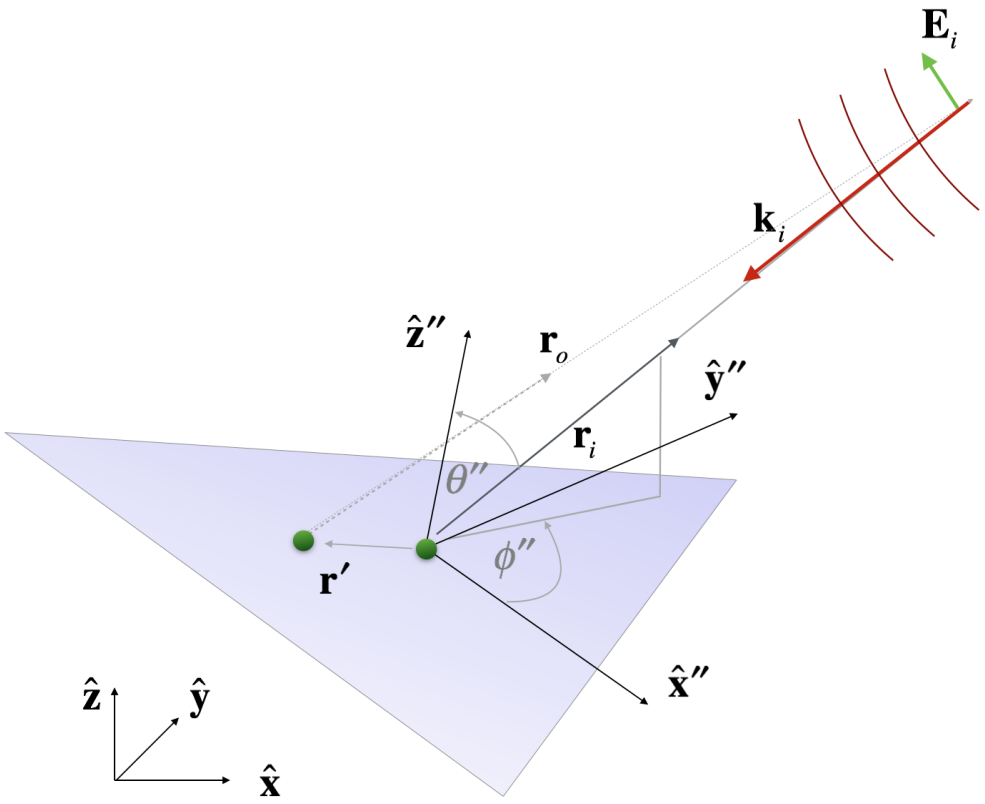


Figure 2.15: Simple Coordinate System for Calculating the Phase of a Scattering Point

The location of the incident ray's hit-point can be used as a reference for the EM field calculations and becomes the origin for subsequent analysis. The EM wave incident on the surface has the form:

$$Ae^{j\vec{k}_i\cdot\vec{r}} \tag{2.314}$$

and so the EM field at scattering location r_0 is provided by:

$$\vec{E}_i = \left(E_{i\theta}\hat{\theta} + E_{i\phi}\hat{\phi}\right)e^{j\vec{k}_i\cdot\vec{r}_o} \tag{2.315}$$

The incident ray is only a representation of the incident EM field which illuminates all parts on the surface of the facet. The incident ray path length to any location on the facet is therefore:

$$\vec{r}_o = \vec{r}_i - \vec{r}' \tag{2.316}$$

Also the incident wavenumber can be considered in terms of the incident path length:

$$\vec{k}_i = k\hat{k}_i = -k\hat{r}_i \tag{2.317}$$

Using these together provides the incident E field for any point on the surface $\vec{r'} = x'\hat{\vec{x}} + y'\hat{\vec{y}} + z'\hat{\vec{z}}$:

$$\vec{E}_i = \left(E_{i\theta}\hat{\theta} + E_{i\phi}\hat{\phi}\right) e^{jk\vec{\hat{r}}_i\cdot\left(\vec{r}_i - \vec{r'}\right)} \tag{2.318}$$

$$\vec{E}_i = \left(E_{i\theta}\hat{\theta} + E_{i\phi}\hat{\phi}\right) e^{jk\left(\vec{\hat{r}}_i\cdot\vec{r}_i - \vec{\hat{r}}_i\cdot\vec{r'}\right)} \tag{2.319}$$

$$\vec{E}_i = \left(E_{i\theta}\hat{\theta} + E_{i\phi}\hat{\phi}\right) e^{-jkr} e^{jk\vec{\hat{r}}_i\cdot\vec{r'}} \tag{2.320}$$

with

$$e^{j\vec{k}_i\cdot\vec{r}_o} = e^{\left(-jkr_i + jk\vec{\hat{r}}_i\cdot\vec{r'}\right)} = e^{-jkr_i} e^{jk\vec{\hat{r}}_i\cdot\vec{r'}} = e^{-jkr_i} e^{jkg} \tag{2.321}$$

and

$$g = \vec{r'}\cdot\vec{\hat{r}}_i \tag{2.322}$$

2.3.1 Electric Surface Currents

Equation (2.320) provided the incident electric fields in terms of the parallel and perpendicular E-field components:

$$\vec{E}_i = \left(E_{i\theta}\hat{\theta} + E_{i\phi}\hat{\phi}\right) e^{-jkr_i} e^{jk\vec{\hat{r}}_i\cdot\vec{r'}} \tag{2.323}$$

To determine the total surface current induced by this incident field, this equation now needs to be deconstructed in terms of the boundary conditions to free up the surface current term. In order to make the analysis simpler a coordinate translation from the global coordinate system into the coordinate system of the triangle is performed as discussed earlier. The analysis starts by defining the direction cosines for the incident ray as being:

$$\begin{aligned} u_i &= \sin\theta_i \cos\phi_i \\ v_i &= \sin\theta_i \sin\phi_i \\ w_i &= \cos\theta_i \end{aligned} \tag{2.324}$$

and so

$$jk\vec{r'}\cdot\vec{\hat{r}}_i = jk\left[x'\sin\theta_i\cos\phi_i + y'\sin\theta_i\sin\phi_i + z'\cos\theta_i\right] \tag{2.325}$$

such that

$$jk\vec{r'}\cdot\vec{\hat{r}}_i = jk\left[x'u_i + y'v_i + z'w_i\right] \equiv jkg \tag{2.326}$$

with

$$g = \vec{r'}\cdot\vec{\hat{r}}_i = \left[x'u_i + y'v_i + z'w_i\right] \tag{2.327}$$

Also for a point on the surface, the normal is defined as being the same as the z axis and so the boundary condition in (2.192) is now

$$\hat{z}'' \times \left[\hat{z}'' \times \vec{E}_{\pm}\right] = -R_s\vec{J}_s \tag{2.328}$$

The final step in setting up the derivation comes from recognising that the transmission fields on the negative side of the sheet can be constructed from the incident ray and the transmission coefficients for both polarisations:

$$\vec{E}_{-} = \vec{E}^t = \left(\hat{\theta}_i T^e_{\parallel} E^i_{\theta} + \hat{\phi}_i T^e_{\perp} E^i_{\phi}\right) e^{j\vec{k}_i\cdot\vec{r}_o} \tag{2.329}$$

Combining equations (2.328) and (2.329) provides:

$$\hat{z}'' \times \left[\hat{z}'' \times \left(\hat{\theta}_i T^e_{\parallel} E^i_{\theta} + \hat{\phi}_i T^e_{\perp} E^i_{\phi}\right) e^{j\vec{k}_i\cdot\vec{r}_o}\right] = -R_s\vec{J}_s \tag{2.330}$$

This can now be expanded but it is worth remembering the various cross products:

$$\hat{z}\times\hat{\theta} = -\hat{x}\cos\theta\sin\phi + \hat{y}\cos\theta\cos\phi \tag{2.331}$$

$$\hat{z}\times\hat{\phi} = -\hat{x}\cos\phi - \hat{y}\sin\phi \tag{2.332}$$

$$\hat{z}\times\left(\hat{z}\times\hat{\theta}\right) = -\left(\hat{x}\cos\theta\cos\phi + \hat{y}\cos\theta\sin\phi\right) \tag{2.333}$$

$$\hat{z}\times\left(\hat{z}\times\hat{\phi}\right) = \hat{x}\sin\phi - \hat{y}\cos\phi \tag{2.334}$$

and so:

$$\left[\left(\hat{z}'' \times \hat{z}'' \times \hat{\theta}_i T_{\parallel}^e E_{\theta}^i + \hat{z}'' \times \hat{z}'' \times \hat{\phi}_i T_{\perp}^e E_{\phi}^i\right) e^{j\vec{k}_i \cdot \vec{r}_o}\right] = -R_s \vec{J}_s \tag{2.335}$$

$$\left[\begin{pmatrix} (-\hat{x}'' \cos\theta_i \cos\phi_i - \hat{y}'' \cos\theta_i \sin\phi_i)\, T_{\parallel}^e E_{\theta}^i \\ +\,(\hat{x}'' \sin\phi_i - \hat{y}'' \cos\phi_i)\, T_{\perp}^e E_{\phi}^i \end{pmatrix} e^{j\vec{k}_i \cdot \vec{r}_o}\right] = -R_s \vec{J}_s \tag{2.336}$$

$$\left[\begin{pmatrix} -\hat{x}'' \cos\theta_i \cos\phi_i T_{\parallel}^e E_{\theta}^i - \hat{y}'' \cos\theta_i \sin\phi_i T_{\parallel}^e E_{\theta}^i \\ +\hat{x}'' \sin\phi_i T_{\perp}^e E_{\phi}^i - \hat{y}'' \cos\phi_i T_{\perp}^e E_{\phi}^i \end{pmatrix} e^{j\vec{k}_i \cdot \vec{r}_o}\right] = -R_s \vec{J}_s \tag{2.337}$$

$$\left[\left(-\hat{x}'' \cos\theta_i \cos\phi_i T_{\parallel}^e E_{\theta}^i + \hat{x}'' \sin\phi_i T_{\perp}^e E_{\phi}^i\right) e^{j\vec{k}_i \cdot \vec{r}_o}\right] = -R_s J_x^s \tag{2.338}$$

$$\left[\left(-\hat{y}'' \cos\theta_i \sin\phi_i T_{\parallel}^e E_{\theta}^i - \hat{y}'' \cos\phi_i T_{\perp}^e E_{\phi}^i\right) e^{j\vec{k}_i \cdot \vec{r}_o}\right] = -R_s J_y^s \tag{2.339}$$

which provides an equation that describes the x and y component of the induced electric surface current on the sheet from the incident E field and the transmission coefficients. Using the derivations for $T_{\parallel}$ and $T_{\perp}$ derived earlier (equations (2.272) and (2.288)) this can be simplified to :

$$2\left(\frac{\cos\theta_i \cos\phi_i}{Z_0 \cos\theta_i + 2R_s} E_{\theta}^i - \frac{\sin\phi_i \cos\theta_i}{2R_s \cos\theta_i + Z_0} E_{\phi}^i\right) e^{-jkr_i} e^{jkg} = J_x^s \tag{2.340}$$

and

$$2\left(\frac{\cos\theta_i \sin\phi_i}{Z_0 \cos\theta_i + 2R_s} E_{\theta}^i + \frac{\cos\phi_i \cos\theta_i}{Z_0 + 2R_s \cos\theta_i} E_{\phi}^i\right) e^{-jkr_i} e^{jkg} = J_y^s \tag{2.341}$$

These two equations can now be used to determine the electric surface current on a planar facet caused by an arbitrary polarisation incident electric field. They represent the case where no magnetic current is flowing and therefore the material has the same magnetic permeability as the surrounding medium. This is the case of a purely electrically resistive material, and in such cases these equations are all that are required for EM modelling and prediction and, indeed are all that are usually implemented.

As discussed in section 2.2.3 these are also all that are required to accurately model the electric current on a completely planar surface as the electrically resistive properties become uncoupled from the magnetically conductive properties [87].

In some situations however, it might be required to consider materials that have

a different *magnetic* permeability. These would have a magnetic current density $\vec{J}_m$ with an associated magnetic conductivity R_m. This means that objects that have different magnetic properties than that of the surrounding medium can be modeled providing the object is perfectly planar.

For completeness the magnetic surface currents are now developed.

2.3.2 Magnetic Surface Currents

The magnetic surface current density can now be calculated in a similar way to those for the electric current density. Considering the magnetic field in the same way as the electric field:

$$\vec{H}^i = \left(H^i_\theta \hat{\theta}_i + H^i_\phi \hat{\phi}_i\right) e^{j\vec{k}_i \cdot \vec{r}_o} \tag{2.342}$$

As with the electric current, the analysis starts by defining the direction cosines as being:

$$\begin{aligned} u_i &= \sin\theta_i \cos\phi_i \\ v_i &= \sin\theta_i \sin\phi_i \\ w_i &= \cos\theta_i \end{aligned} \tag{2.343}$$

and by recognising that

$$jk\vec{r'} \cdot \hat{\vec{r}}_i = jk\left[x' \sin\theta_i \cos\phi_i + y' \sin\theta_i \sin\phi_i + z' \cos\theta_i\right] \tag{2.344}$$

such that

$$jk\vec{r'} \cdot \hat{\vec{r}}_i = jk\left[x'u_i + y'v_i + z'w_i\right] \equiv jkg \tag{2.345}$$

with

$$g = \vec{r'} \cdot \hat{\vec{r}}_i = \left[x'u_i + y'v_i + z'w_i\right] \tag{2.346}$$

Also from (2.220):

$$\hat{n} \times \left(\hat{n} \times \vec{H}_\pm\right) = -R_m \vec{M} \tag{2.347}$$

as $\vec{H}_-$ as $\vec{H}_\pm$:

$$\vec{H}_- = \vec{H}^t\Big|_{z''=0} \tag{2.348}$$

$$\vec{H}_- = \vec{H}^t = \left(\hat{\theta}_i T^m_\parallel H^i_\theta + \hat{\phi}_i T^m_\perp H^i_\phi\right) e^{j\vec{k}_i \cdot \vec{r}_o} \tag{2.349}$$

and as $\hat{z}'' = \hat{n}$ then:

$$\hat{z}'' \times \left[\hat{z}'' \times \left(\hat{\theta}_i T_{\parallel}^m H_\theta^i + \hat{\phi}_i T_{\perp}^m H_\phi^i \right) e^{j\vec{k}_i \cdot \vec{r}_o} \right] = -R_m \vec{M} \tag{2.350}$$

Using the standard triple vector products:

$$\hat{z}'' \times \left(\hat{z}'' \times \hat{\theta}_i \right) = -\left(\hat{x}'' \cos\theta_i \cos\phi_i + \hat{y}'' \cos\theta_i \sin\phi_i \right) \tag{2.351}$$

$$\hat{z}'' \times \left(\hat{z}'' \times \hat{\phi}_i \right) = \hat{x}'' \sin\phi_i - \hat{y}'' \cos\phi_i \tag{2.352}$$

The equation becomes

$$\begin{pmatrix} \left(-\hat{x}'' \cos\theta_i \cos\phi_i - \hat{y}'' \cos\theta_i \sin\phi_i \right) T_{\parallel}^m H_\theta^i \\ + \left(\hat{x}'' \sin\phi_i - \hat{y}'' \cos\phi_i \right) T_{\perp}^m H_\phi^i \end{pmatrix} e^{j\vec{k}_i \cdot \vec{r}_o} = -R_m \vec{M} \tag{2.353}$$

This can now be simplified using the terms that we developed for the transmission coefficients in (2.300) and (2.313):

$$\begin{aligned} &\left(-\hat{x}'' \cos\theta_i \cos\phi_i - \hat{y}'' \cos\theta_i \sin\phi_i \right) \left(\frac{2\eta_0 R_m \cos\theta_i}{1 + 2\eta_0 R_m \cos\theta_i} \right) H_\theta^i e^{j\vec{k}_i \cdot \vec{r}_o} \\ &+ \left(\hat{x} \sin\phi_i - \hat{y} \cos\phi_i \right) \left(\frac{2\eta_0 R_m}{\cos\theta_i + 2\eta_0 R_m} \right) H_\phi^i e^{j\vec{k}_i \cdot \vec{r}_o} = -R_m \vec{M} \end{aligned} \tag{2.354}$$

$$2Z_0 \left(\frac{-\cos^2\theta_i \cos\phi_i}{1 + 2Z_0 R_m \cos\theta_i} H_\theta^i + \frac{\sin\phi_i}{\cos\theta_i + 2Z_0 R_m} H_\phi^i \right) e^{-jkr_i} e^{jkg} = -M_x \tag{2.355}$$

$$2Z_0 \left(\frac{\cos^2\theta_i \sin\phi_i}{1 + 2Z_0 R_m \cos\theta_i} H_\theta^i - \frac{\cos\phi_i}{\cos\theta_i + 2Z_0 R_m} H_\phi^i \right) e^{-jkr_i} e^{jkg} = -M_y \tag{2.356}$$

2.3.3 Summary

This section started with a description of material properties and developed from them a set of boundary conditions that relate not only to perfect electrical conductors but also to more realistic materials such as dielectrics. In order to provide the full range of materials available it hypothesised the existence of a magnetically conductive sheet that, when aligned with an electrically resistive sheet allows all materials to be captured in terms of their intrinsic properties. These properties

were then used to develop the boundary conditions for the hypothetical sheet. The boundary conditions were then used to derive, through a set of intermediate steps, the surface currents on the material. The dependence of the surface currents on polarisation was also captured which provides a way to predict the polarimetric effects of the incident radiation on the material.

The surface currents can now be used with the far field scattering equations derived in the previous section. There is however a problem hiding in the form of those initial equations: the double integral term in equations (2.151)-(2.154). This is not an easy challenge to overcome and will be discussed in the next section.

2.4 Scattered Fields From a Planar Surface

2.4.1 Introduction

The previous section presented a way to derive the electric and magnetic surface currents J_x^s, J_y^s, J_x^m and J_y^m that are induced on a scattering surface by an incident EM wave. The electric surface currents are:

$$2\left(\frac{\cos\theta_i\cos\phi_i}{Z_0\cos\theta_i+2R_s}E_\theta^i-\frac{\sin\phi_i\cos\theta_i}{Z_0+2R_s\cos\theta_i}E_\phi^i\right)e^{-jkr_i}e^{jkg}=J_x^s \tag{2.357}$$

and

$$2\left(\frac{\cos\theta_i\sin\phi_i}{Z_0\cos\theta_i+2R_s}E_\theta^i+\frac{\cos\phi_i\cos\theta_i}{Z_0+2R_s\cos\theta_i}E_\phi^i\right)e^{-jkr_i}e^{jkg}=J_y^s \tag{2.358}$$

The magnetic surface currents are given by:

$$2Z_0\left(\frac{-\cos^2\theta_i\cos\phi_i}{1+2Z_0R_m\cos\theta_i}H_\theta^i+\frac{\sin\phi_i}{\cos\theta_i+2Z_0R_m}H_\phi^i\right)e^{-jkr_i}e^{jkg}=-M_x \tag{2.359}$$

and

$$2Z_0\left(\frac{\cos^2\theta_i\sin\phi_i}{1+2Z_0R_m\cos\theta_i}H_\theta^i-\frac{\cos\phi_i}{\cos\theta_i+2Z_0R_m}H_\phi^i\right)e^{-jkr_i}e^{jkg}=-M_y \tag{2.360}$$

These can now be used in the far-field scattering equations derived in 2.1 which are (for reference):

$$E_r\simeq 0 \tag{2.361}$$

$$E_\theta \simeq -jk\frac{e^{-jkr}}{4\pi r}\left(L_\phi + Z_0 N_\theta\right) \tag{2.362}$$

$$E_\phi \simeq +\frac{jke^{-jkr}}{4\pi r}\left(L_\theta - Z_0 N_\phi\right) \tag{2.363}$$

$$H_r \simeq 0 \tag{2.364}$$

$$H_\theta \simeq +\frac{jke^{-jkr}}{4\pi r}\left(N_\phi - \frac{L_\theta}{Z_0}\right) \tag{2.365}$$

$$H_\phi \simeq -\frac{jke^{-jkr}}{4\pi r}\left(N_\theta + \frac{L_\phi}{Z_0}\right) \tag{2.366}$$

with

$$N_\theta = \iint_s \left(J_x\cos\theta\cos\phi + J_y\cos\theta\sin\phi - J_z\sin\theta\right)e^{jkr'\cos\psi_s}ds' \tag{2.367}$$

$$N_\phi = \iint_s \left(-J_x\sin\phi + J_y\cos\phi\right)e^{jkr'\cos\psi_s}ds' \tag{2.368}$$

$$L_\theta = \iint_s \left(M_x\cos\theta\cos\phi + M_y\cos\theta\sin\phi - M_z\sin\theta\right)e^{jkr'\cos\psi_s}ds' \tag{2.369}$$

$$L_\phi = \iint_s \left(-M_x\sin\phi + M_y\cos\phi\right)e^{jkr'\cos\psi_s}ds' \tag{2.370}$$

A challenge now arises in solving the surface integral in equations (2.367) through (2.370). This will be the subject of this section.

The approach will be to first apply some assumptions that simplify the task and demonstrate how a modest solution can be developed if the surface facets are comprised of rectangular tiles. The second part of this section will be to look at the more complicated situation of triangular facets.

2.4.2 Parallel Incident Polarisation

When the field is parallel polarised then $E^i_\phi = 0$ and so equations (2.357) and (2.358) become:

$$J_x = \frac{2\cos\theta_i\cos\phi_i}{Z_0\cos\theta_i + 2R_s}E^i_\theta e^{-jkr_i}e^{jkg} \tag{2.371}$$

$$J_y = \frac{2\cos\theta_i\sin\phi_i}{Z_0\cos\theta_i + 2R_s}E^i_\theta e^{-jkr_i}e^{jkg} \tag{2.372}$$

Also $H_\theta^i = 0$ and so equations (2.359) and (2.360) become:

$$M_x = -\frac{2Z_0 \sin\phi_i}{\cos\theta_i + 2Z_0 R_m} H_\phi^i e^{-jkr_i} e^{jkg} \tag{2.373}$$

$$M_y = \frac{2Z_0 \cos\phi_i}{\cos\theta_i + 2Z_0 R_m} H_\phi^i e^{-jkr_i} e^{jkg} \tag{2.374}$$

In order to use these equations with (2.367) the $\hat{z}$ term can be removed by first rotating the global coordinates into the local coordinate plane using (2.217) and the double prime terminology to represent angles in the facet plane to provide:

$$N_\theta = \iint_s \left(\begin{array}{l} \left(\dfrac{2\cos\theta_i \cos\phi_i}{Z_0 \cos\theta_i + 2R_s} E_\theta^i e^{-jkr_i} e^{jkg} \right) \cos\theta'' \cos\phi'' \\ + \left(\dfrac{2\cos\theta_i \sin\phi_i}{Z_0 \cos\theta_i + 2R_s} E_\theta^i e^{-jkr_i} e^{jkg} \right) \cos\theta'' \sin\phi'' \end{array} \right) e^{jkr' \cos\psi_s} ds' \tag{2.375}$$

which can be simplified by moving the static parameters outside the integral:

$$N_\theta = 2E_\theta^i \cos\theta_i \cos\theta'' \cos(\phi'' - \phi_i) \iint_s \left(\frac{e^{-jkr_i} e^{jkg} e^{jkr' \cos\psi_s}}{Z_0 \cos\theta_i + 2R_s} \right) dx'dy' \tag{2.376}$$

Equally N_ϕ becomes (from (2.368)):

$$N_\phi = \iint_s \left(\begin{array}{l} -\dfrac{2\cos\theta_i \cos\phi_i}{Z_0 \cos\theta_i + 2R_s} E_\theta^i e^{-jkr_i} e^{jkg} \sin\phi'' \\ + \dfrac{2\cos\theta_i \sin\phi_i}{Z_0 \cos\theta_i + 2R_s} E_\theta^i e^{-jkr_i} e^{jkg} \cos\phi'' \end{array} \right) e^{jkr' \cos\psi_s} ds' \tag{2.377}$$

$$N_\phi = 2E_\theta^i \cos\theta_i \sin(\phi_i - \phi'') \iint_s \frac{e^{-jkr_i} e^{jkg} e^{jkr' \cos\psi_s}}{Z_0 \cos\theta_i + 2R_s} dx'dy' \tag{2.378}$$

In a similar way we can rewrite terms for L_θ and L_ϕ :

$$L_\theta = 2Z_0 \cos\theta'' \sin(\phi'' - \phi_i) H_\phi^i \iint_s \frac{e^{-jkr_i} e^{jkg} e^{jkr' \cos\psi_s}}{\cos\theta_i + 2Z_0 R_m} dx'dy' \tag{2.379}$$

$$L_\phi = 2Z_0 \cos(\phi_i - \phi'') H_\phi^i \iint_s \frac{e^{-jkr_i} e^{jkg} e^{jkr' \cos\psi_s}}{\cos\theta_i + 2Z_0 R_m} dx'dy' \tag{2.380}$$

In the real world R_s and R_m would vary over the surface of the target object.

However if the target facet is selected to be small enough that the resistivity / conductivity of the material hardly changes then they can be removed from the integral. Equally if the facet is small enough and/or far enough away from the source of illumination then $\hat{r}_i$ is constant over the facet and can also be taken outside of the integrals to provide:

$$N_\theta = \frac{2\cos\theta_i \cos\theta'' \cos(\phi'' - \phi_i) E^i_\theta}{Z_0 \cos\theta_i + 2R_s} e^{-jkr_i} \iint_s e^{jk(g+h)} dx'dy' \tag{2.381}$$

$$N_\phi = \frac{2\cos\theta_i \sin(\phi_i - \phi'') E^i_\theta}{Z_0 \cos\theta_i + 2R_s} e^{-jkr_i} \iint_s e^{jk(g+h)} dx'dy' \tag{2.382}$$

$$L_\theta = \frac{2Z_0 \cos\theta'' \sin(\phi'' - \phi_i) H^i_\phi}{\cos\theta_i + 2Z_0 R_m} e^{-jkr_i} \iint_s e^{jk(g+h)} dx'dy' \tag{2.383}$$

$$L_\phi = \frac{2Z_0 \cos(\phi_i - \phi'') H^i_\phi}{\cos\theta_i + 2Z_0 R_m} e^{-jkr_i} \iint_s e^{jk(g+h)} dx'dy' \tag{2.384}$$

Where here h has been defined for convenience as:

$$h = r' \cos\psi_s \tag{2.385}$$

These equations represent the specific condition when the incident EM field is polarised parallel to the plane of incidence.

2.4.3 Perpendicular Incident Polarisation

When the incident field is perpendicularly polarised then $E^i_\theta = 0$ and $H^i_\phi = 0$ equations (2.357) through (2.360) therefore become:

$$J_x = -\frac{2\sin\phi_i \cos\theta_i}{Z_0 + 2R_s \cos\theta_i} E^i_\phi e^{-jkr_i} e^{jkg} \tag{2.386}$$

$$J_y = \frac{2\cos\phi_i \cos\theta_i}{Z_0 + 2R_s \cos\theta_i} E^i_\phi e^{-jkr_i} e^{jkg} \tag{2.387}$$

$$M_x = \frac{2Z_0 \cos^2\theta_i \cos\phi_i}{1 + 2Z_0 R_m \cos\theta_i} H^i_\theta e^{-jkr_i} e^{jkg} \tag{2.388}$$

$$M_y = -\frac{2Z_0\cos^2\theta_i \sin\phi_i}{1+2Z_0R_m\cos\theta_i} H_\theta^i e^{-jkr_i} e^{jkg} \tag{2.389}$$

These equations can now be combined with equations (2.151) through (2.154) to become (again noting the coordinate rotation into the plane of the facet to remove the $\hat{z}$ term):

$$N_\theta = 2\cos\theta_i \cos\theta'' \sin(\phi'' - \phi_i) E_\phi^i \iint_s \frac{e^{-jkr_i} e^{jkg} e^{jkr'\cos\psi_s}}{Z_0 + 2R_s\cos\theta_i} dx'dy' \tag{2.390}$$

$$N_\phi = 2\cos\theta_i \cos(\phi_i - \phi'') E_\phi^i \iint_s \frac{e^{-jkr_i} e^{jkg} e^{jkr'\cos\psi_s}}{Z_0 + 2R_s\cos\theta_i} dx'dy' \tag{2.391}$$

$$L_\theta = 2Z_0\cos^2\theta_i \cos\theta'' \cos(\phi_i + \phi'') H_\theta^i \iint_s \frac{e^{-jkr_i} e^{jkg} e^{jkr'\cos\psi_s}}{1+2Z_0R_m\cos\theta_i} dx'dy' \tag{2.392}$$

$$L_\phi = -2Z_0\cos^2\theta_i \sin(\phi'' + \phi_i) H_\theta^i \iint_s \frac{e^{-jkr_i} e^{jkg} e^{jkr'\cos\psi_s}}{1+2Z_0R_m\cos\theta_i} dx'dy' \tag{2.393}$$

Assuming the surface facet is small enough that the resistivity and conductivity of the material doesn't change over its surface and that the incident rays are planar (i.e. the direction to the source $\hat{r}_i$ does not change) then we can take terms out of the integration:

$$N_\theta = \frac{2\cos\theta_i \cos\theta'' \sin(\phi'' - \phi_i)}{Z_0 + 2R_s\cos\theta_i} E_\phi^i e^{-jkr_i} \iint_s e^{jk(g+h)} dx'dy' \tag{2.394}$$

$$N_\phi = \frac{2\cos\theta_i \cos(\phi_i - \phi'')}{Z_0 + 2R_s\cos\theta_i} E_\phi^i e^{-jkr_i} \iint_s e^{jk(g+h)} dx'dy' \tag{2.395}$$

$$L_\theta = \frac{2Z_0\cos^2\theta_i \cos\theta'' \cos(\phi_i + \phi'')}{1+2Z_0R_m\cos\theta_i} H_\theta^i e^{-jkr_i} \iint_s e^{jk(g+h)} dx'dy' \tag{2.396}$$

$$L_\phi = -\frac{2Z_0\cos^2\theta_i \sin(\phi'' + \phi_i)}{1+2Z_0R_m\cos\theta_i} H_\theta^i e^{-jkr_i} \iint_s e^{jk(g+h)} dx'dy' \tag{2.397}$$

where again we have used $h = r'\cos\psi_s$. These terms represent the specific condition when the incident EM field is polarised perpendicular to the plane of incidence. They will now be used to explore the scattering from facet primitives.

2.4.4 The E-Field for a Rectangular Facet

Having calculated the equations for N and L they can now be inserted into (2.361) through (2.366) to calculate the scattered fields at the receiver. Its worth remembering that the derivations so far have assumed that the facet is completely planar and that the fields at the observation point can be constructed through the integration of surface currents on the facet. To make the derivation easier the problem will once again be decomposed into parallel and perpendicular polarisations and then combined at the end. The analysis starts with parallel polarisation.

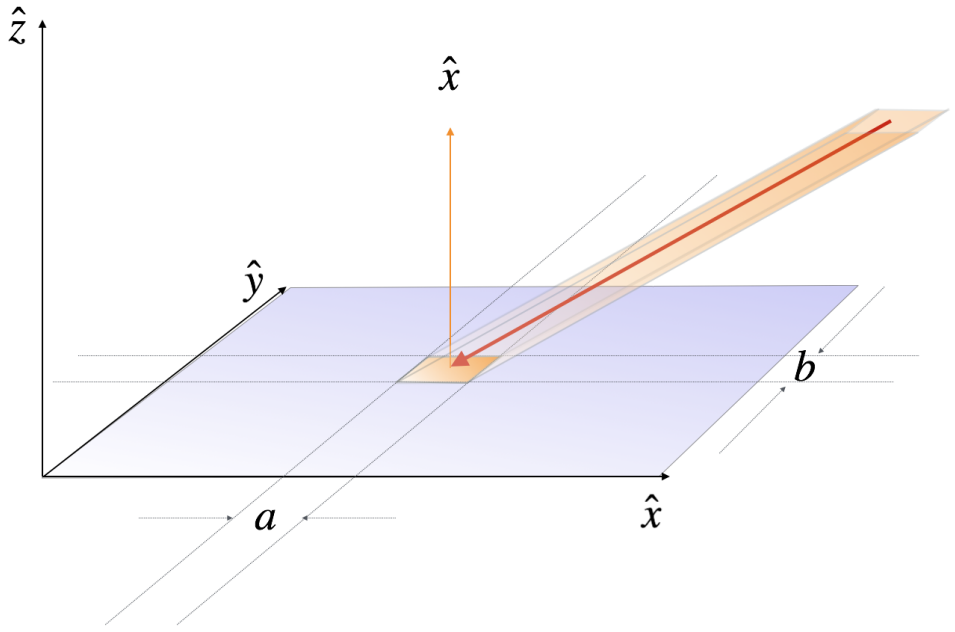


Figure 2.16: EM Ray Incident on a Planar Surface

Consider a sheet illuminated by a series of rays. Each ray could be constructed as rectangular tubes such that their footprint on the surface of the sheet has a rectangular area of ab (Figure 2.16). In such a situation it would be possible to derive a closed form solution to the scattered E-fields created by the induced currents on the rectangle.

2.4.5 Parallel Polarisation Case

For the case in Figure 2.16 when the incident E field is polarised parallel to the plane of incidence the N and L coefficients become:

$$N_\theta = \frac{2\cos\theta_i \cos\theta'' \cos(\phi'' - \phi_i) E_\theta^i}{Z_0 \cos\theta_i + 2R_s} e^{-jkr_i} \int_{-b/2}^{b/2} \int_{-a/2}^{a/2} e^{jk(g+h)} dx' dy' \tag{2.398}$$

$$N_\phi = \frac{2\cos\theta_i \sin(\phi_i - \phi'') E_\theta^i}{Z_0 \cos\theta_i + 2R_s} e^{-jkr_i} \int_{-b/2}^{b/2} \int_{-a/2}^{a/2} e^{jk(g+h)} dx' dy' \tag{2.399}$$

$$L_\theta = \frac{2Z_0 \cos\theta'' \sin(\phi'' - \phi_i) H_\phi^i}{\cos\theta_i + 2Z_0 R_m} e^{-jkr_i} \int_{-b/2}^{b/2} \int_{-a/2}^{a/2} e^{jk(g+h)} dx' dy' \tag{2.400}$$

$$L_\phi = \frac{2Z_0 \cos(\phi_i - \phi'') H_\phi^i}{\cos\theta_i + 2Z_0 R_m} e^{-jkr_i} \int_{-b/2}^{b/2} \int_{-a/2}^{a/2} e^{jk(g+h)} dx' dy' \tag{2.401}$$

Using the integral identity:

$$\int_{-c/2}^{c/2} e^{j\alpha z} dz = c \left[\frac{\sin\left(\frac{\alpha}{2} c\right)}{\frac{\alpha}{2} c} \right] \tag{2.402}$$

The terms become:

$$N_\theta = \frac{2E_\theta^i \cos\theta_i \cos\theta'' \cos(\phi'' - \phi_i) e^{-jkr_i}}{Z_0 \cos\theta_i + 2R_s} ab \frac{\sin(X)}{X} \frac{\sin(Y)}{Y} \tag{2.403}$$

$$N_\phi = \frac{2E_\theta^i \cos\theta_i \sin(\phi_i - \phi'') e^{-jkr_i}}{Z_0 \cos\theta_i + 2R_s} ab \frac{\sin(X)}{X} \frac{\sin(Y)}{Y} \tag{2.404}$$

$$L_\theta = \frac{2Z_0 H_\phi^i \cos\theta'' \sin(\phi'' - \phi_i) e^{-jkr_i}}{\cos\theta_i + 2Z_0 R_m} ab \frac{\sin(X)}{X} \frac{\sin(Y)}{Y} \tag{2.405}$$

$$L_\phi = \frac{2Z_0 H_\phi^i \cos(\phi_i - \phi'') e^{-jkr_i}}{\cos\theta_i + 2Z_0 R_m} ab \frac{\sin(X)}{X} \frac{\sin(Y)}{Y} \tag{2.406}$$

with

$$X = \frac{k\left(h_x + h_x\right)}{2}a \qquad Y = \frac{k\left(g_y + h_y\right)}{2}b \tag{2.407}$$

It is therefore possible to reconstruct the total scattered field from equations (2.362) through (2.366):

$$E_{\theta\theta} = -\frac{jZ_0 k e^{-jkr} e^{-jkr_i}}{2\pi r} ab \frac{\sin(X)}{X}\frac{\sin(Y)}{Y}\left(\begin{array}{l}\dfrac{\cos\theta_i \cos\theta'' \cos(\phi'' - \phi_i)}{Z_0 \cos\theta_i + 2R_s} E_\theta^i \\ +\dfrac{\cos(\phi_i - \phi'')}{\cos\theta_i + 2Z_0 R_m} H_\phi^i\end{array}\right) \tag{2.408}$$

$$E_{\phi\theta} = \frac{jZ_0 k e^{-jkr} e^{-jkr_i}}{2\pi r} ab \frac{\sin(X)}{X}\frac{\sin(Y)}{Y}\left(\begin{array}{l}\dfrac{\cos\theta'' \sin(\phi'' - \phi_i)}{\cos\theta_i + 2Z_0 R_m} H_\phi^i \\ -\dfrac{\cos\theta_i \sin(\phi_i - \phi'')}{Z_0 \cos\theta_i + 2R_s} E_\theta^i\end{array}\right) \tag{2.409}$$

$$H_{\theta\theta} = \frac{jk e^{-jkr} e^{-jkr_i}}{2\pi r} ab \frac{\sin(X)}{X}\frac{\sin(Y)}{Y}\left(\begin{array}{l}\dfrac{\cos\theta_i \sin(\phi_i - \phi'')}{Z_0 \cos\theta_i + 2R_s} E_\theta^i \\ -\dfrac{\cos\theta'' \sin(\phi'' - \phi_i)}{\cos\theta_i + 2Z_0 R_m} H_\phi^i\end{array}\right) \tag{2.410}$$

$$H_{\phi\theta} = -\frac{jk e^{-jkr} e^{-jkr_i}}{2\pi r} ab \frac{\sin(X)}{X}\frac{\sin(Y)}{Y}\left(\begin{array}{l}\dfrac{\cos\theta_i \cos\theta'' \cos(\phi'' - \phi_i)}{Z_0 \cos\theta_i + 2R_s} E_\theta^i \\ +\dfrac{\cos(\phi_i - \phi'')}{\cos\theta_i + 2Z_0 R_m} H_\phi^i\end{array}\right) \tag{2.411}$$

where the first subscript denotes the polarisation of the scattered field and the second subscript denotes the polarisation of the incident field.

2.4.6 Perpendicular Polarisation Case

For the case in Figure 2.16 where the incident ray is polarised perpendicular to the plane of incidence, the N and L coefficients become:

$$N_\theta = \frac{2E_\phi^i \cos\theta_i \cos\theta'' \sin(\phi'' - \phi_i) e^{-jkr_i}}{Z_0 + 2R_s \cos\theta_i} \int_{-b/2}^{b/2}\int_{-a/2}^{a/2} e^{jk(g+h)} dx' dy' \tag{2.412}$$

$$N_\phi = \frac{2E_\phi^i \cos\theta_i \cos(\phi_i - \phi'') e^{-jkr_i}}{Z_0 + 2R_s \cos\theta_i} \int_{-b/2}^{b/2}\int_{-a/2}^{a/2} e^{jk(g+h)} dx' dy' \tag{2.413}$$

$$L_\theta = \frac{2Z_0 H_\theta^i \cos^2\theta_i \cos\theta'' \cos(\phi_i + \phi'')\, e^{-jkr_i}}{1 + 2Z_0 R_m \cos\theta_i} \int_{-b/2}^{b/2}\int_{-a/2}^{a/2} e^{jk(g+h)} dx' dy' \tag{2.414}$$

$$L_\phi = -\frac{2Z_0 H_\theta^i \cos^2\theta_i \sin(\phi'' + \phi_i)\, e^{-jkr_i}}{1 + 2Z_0 R_m \cos\theta_i} \int_{-b/2}^{b/2}\int_{-a/2}^{a/2} e^{jk(g+h)} dx' dy' \tag{2.415}$$

Using the integral identity in (2.402) the terms become:

$$N_\theta = \frac{2E_\phi^i \cos\theta_i \cos\theta'' \sin(\phi'' - \phi_i)\, e^{-jkr_i}}{Z_0 + 2R_s \cos\theta_i} ab \left[\frac{\sin(X)}{X}\right]\left[\frac{\sin(Y)}{Y}\right] \tag{2.416}$$

$$N_\phi = \frac{2E_\phi^i \cos\theta_i \cos(\phi_i - \phi'')\, e^{-jkr_i}}{Z_0 + 2R_s \cos\theta_i} ab \left[\frac{\sin(X)}{X}\right]\left[\frac{\sin(Y)}{Y}\right] \tag{2.417}$$

$$L_\theta = \frac{2Z_0 H_\theta^i \cos^2\theta_i \cos\theta'' \cos(\phi_i + \phi'')\, e^{-jkr_i}}{1 + 2Z_0 R_m \cos\theta_i} ab \left[\frac{\sin(X)}{X}\right]\left[\frac{\sin(Y)}{Y}\right] \tag{2.418}$$

$$L_\phi = -\frac{2Z_0 H_\theta^i \cos^2\theta_i \sin(\phi'' + \phi_i)\, e^{-jkr_i}}{1 + 2Z_0 R_m \cos\theta_i} ab \left[\frac{\sin(X)}{X}\right]\left[\frac{\sin(Y)}{Y}\right] \tag{2.419}$$

with X and Y defined as (2.407). Again it is possible to reconstruct the total scattered field from equations (2.362) through (2.366):

$$E_{\theta\phi} = \frac{jZ_0 k e^{-jkr} e^{-jkr_i}}{2\pi r} ab \frac{\sin(X)}{X}\frac{\sin(Y)}{Y} \begin{pmatrix} \dfrac{H_\theta^i \cos^2\theta_i \sin(\phi'' + \phi_i)}{1 + 2Z_0 R_m \cos\theta_i} \\ -\dfrac{E_\phi^i \cos\theta_i \cos\theta'' \sin(\phi'' - \phi_i)}{Z_0 + 2R_s \cos\theta_i} \end{pmatrix} \tag{2.420}$$

$$E_{\phi\phi} = \frac{jZ_0 k e^{-jkr} e^{-jkr_i}}{2\pi r} ab \frac{\sin(X)}{X}\frac{\sin(Y)}{Y} \begin{pmatrix} \dfrac{H_\theta^i \cos^2\theta_i \cos\theta'' \cos(\phi_i + \phi'')}{1 + 2Z_0 R_m \cos\theta_i} \\ -\dfrac{E_\phi^i \cos\theta_i \cos(\phi_i - \phi'')}{Z_0 + 2R_s \cos\theta_i} \end{pmatrix} \tag{2.421}$$

$$H_{\theta\phi} = \frac{jk e^{-jkr} e^{-jkr_i}}{2\pi r} ab \left[\frac{\sin(X)}{X}\right]\left[\frac{\sin(Y)}{Y}\right] \begin{pmatrix} \dfrac{\cos\theta_i \cos(\phi_i - \phi'')}{Z_0 + 2R_s \cos\theta_i} E_\phi^i \\ -\dfrac{\cos^2\theta_i \cos\theta'' \cos(\phi_i + \phi'')}{1 + 2Z_0 R_m \cos\theta_i} H_\theta^i \end{pmatrix} \tag{2.422}$$

$$H_{\phi\phi} = -\frac{jke^{-jkr}e^{-jkr_i}}{2\pi r}ab\left[\frac{\sin(X)}{X}\right]\left[\frac{\sin(Y)}{Y}\right]\begin{pmatrix}\dfrac{\cos\theta_i\cos\theta''\sin(\phi''-\phi_i)}{Z_0+2R_s\cos\theta_i}E_\phi^i \\ -\dfrac{\cos^2\theta_i\sin(\phi''+\phi_i)}{1+2Z_0R_m\cos\theta_i}H_\theta^i\end{pmatrix} \tag{2.423}$$

and again the first subscript denotes the scattered polarisation and the second subscript denotes the incident polarisation.

2.4.7 Arbitrary Incident Polarisation

In general the incident wave is composed on both parallel and perpendicular components. The total polarisation components of the scattered field are given by:

$$E_\theta^s = E_{\theta\theta}^s + E_{\theta\phi}^s \tag{2.424}$$

and

$$E_\phi^s = E_{\phi\theta}^s + E_{\phi\phi}^s \tag{2.425}$$

and so

$$E_{\theta\theta} = C_0\left(-\frac{\cos\theta_i\cos\theta''\cos(\phi''-\phi_i)}{Z_0\cos\theta_i+2R_s}E_\theta^i - \frac{\cos(\phi_i-\phi'')}{\cos\theta_i+2Z_0R_m}H_\phi^i\right) \tag{2.426}$$

$$E_{\phi\theta} = C_0\left(\frac{\cos\theta''\sin(\phi''-\phi_i)}{\cos\theta_i+2Z_0R_m}H_\phi^i - \frac{\cos\theta_i\sin(\phi_i-\phi'')}{Z_0\cos\theta_i+2R_s}E_\theta^i\right) \tag{2.427}$$

$$E_{\theta\phi} = C_0\left(\frac{\cos^2\theta_i\sin(\phi''+\phi_i)}{1+2Z_0R_m\cos\theta_i}H_\theta^i - \frac{\cos\theta_i\cos\theta''\sin(\phi''-\phi_i)}{Z_0+2R_s\cos\theta_i}E_\phi^i\right) \tag{2.428}$$

$$E_{\phi\phi} = C_0\left(\frac{\cos^2\theta_i\cos\theta''\cos(\phi_i+\phi)}{1+2Z_0R_m\cos\theta_i}H_\theta^i - \frac{\cos\theta_i\cos(\phi_i-\phi'')}{Z_0+2R_s\cos\theta_i}E_\phi^i\right) \tag{2.429}$$

where

$$C_0 = \frac{jZ_0ke^{-jkr}e^{-jkr_i}}{2\pi r}ab\frac{\sin(X)}{X}\frac{\sin(Y)}{Y} \tag{2.430}$$

This approach is convenient and simple to implement and can provide good results with careful implementation [89]. It does however have a few limitations. Firstly there is a need to make sure that all the rays are contiguous within the transmitted radar beam. This means that for fewer rays, the area of each ray *ab* is larger. This becomes a problem near the edges of an object where the ray-

rectangle overlaps the edge of the facet and results in a reduced radar cross section. There are approaches to address this (e.g. using variable sized rectangles) but the implementation is always tricky and the solutions are often more computationally intensive.

It is therefore worth considering a second approach - that of integrating the fields over a triangle.

2.4.8 The E-Field for a Triangular Facet

Triangular facets are useful for CAD modelling as they can easily be used to create any three dimensional surface. If the triangles are very small and/or allowed to be curved then accurate CAD modelling and ray tracing can be implemented. As this work is concerned with the implementation of the combined electrically resistive and magnetically conductive materials, it is essential that the triangles are planar and so curved triangle facets are not considered.

The approach presented here was first developed by A.C. Ludwig [90] in 1968 as a mathematical technique to solve the field integral for EM scattering and diffraction patterns over a rectangular surface. The problem is really one of computational efficiency in trying to solve an equation of the form:

$$I = \iint_S F(\theta,\phi)\, e^{-kf(\theta,\phi)} d\theta, d\phi \tag{2.431}$$

where $k = 2\pi/\lambda$ and F and f are two functions that vary with spherical coordinates θ, ϕ. The form of this equation can be seen in equations (2.151) through (2.154) and is at the heart of solving the surface integral over a facet. The issue comes in recognising that the exponential term in the equation can undergo many 2π cycles over the surface of the facet. As a result, conventional integration simplifications such as Simpson's rule are not adequate to provide a solution in any meaningful timescale. The only solution to this is to split the facet up into sub-facets, each a fraction of the wavelength and then to apply Simpson's rule to the individual elements before summing the sub-facets together. This is clearly a computational burden, especially for larger scenes with many facets much greater than the wavelength

dimension[3].

Ludwig's approach was to treat the amplitude and phase as two independent linear functions (Figure 2.17) and to perform the integration across each of them before combining the results.

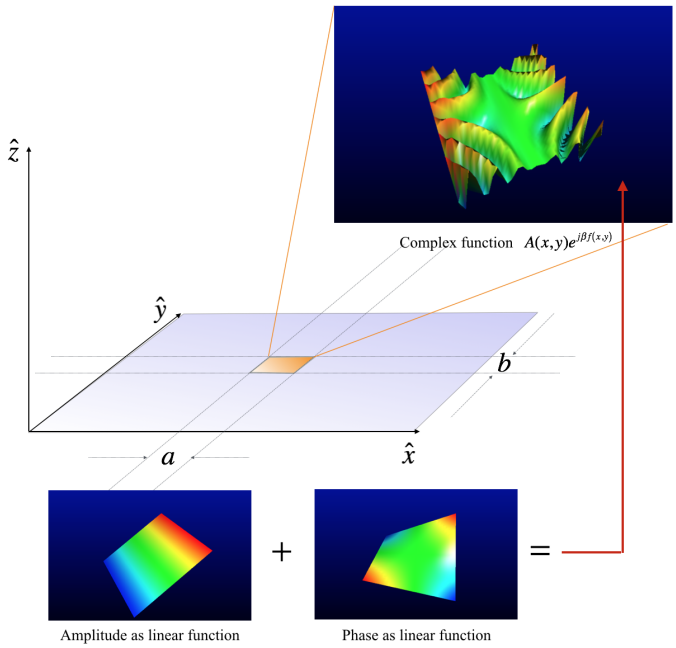


Figure 2.17: Ludwig's integration approximation deals with the summation of a frequently varying complex function by integrating amplitude and phase individually as linear functions

This technique was later adapted up by Pogorzelski [91] in 1985 to work with triangular facets which have a significant benefit to CAD modelling over the use of square or rectangular tiles. A further enhancement to this approach was developed by Dos Santos [92] to work with barycentric coordinate systems which offer significant performance increases for ray-facet intersection and will be discussed in Chapter 3. A final stage in the development of the technique was then implemented by Fernando Moreira and Aluizio Prata [93] in 1994 who demonstrated that the previous work could be combined with an error-checker to tessellate many triangles to model the surface of antennas. The approach presented here was used by Garrido [94] and Chatzigeorgiadis [95] from the US Naval Postgraduate School in Monterey for

[3]It is interesting to note however that in 1968 Ludwig's definition of computational burden was performing an integration grid over 50x181 samples and storing the results as 54,000 'words' (complex vectors). In 2018 the same calculation can be performed on a watch!

physical optics (PO) radar cross section prediction.

Using the terminology used by [93] the integral equation for the surface currents can be rewritten as:

$$I_c = \iint_S C(\eta,\xi)\, e^{jD(\eta,\xi)} d\eta, d\xi \tag{2.432}$$

where

$$C(\eta,\xi) = C_p\eta + C_q\xi + C_0 \equiv \left|\vec{E}_i\right| \tag{2.433}$$

and represents the amplitude function specified with barycentric coordinates η,ξ. Setting the amplitude of the incident wave to be *1* simplifies the analysis:

$$\left|\vec{E}_i\right| = 1 \Rightarrow C_p = C_q = 0,\ C_0 = 1 \tag{2.434}$$

The phase function is then represented by the barycentric coordinates:

$$D(\eta,\xi) = D_p\eta + D_q\xi + D_0 \tag{2.435}$$

where

$$\begin{aligned} D_p &= k\left[(x_1 - x_3)u + (y_1 - y_3)v + (z_1 - z_3)w\right] \\ D_q &= k\left[(x_2 - x_3)u + (y_2 - y_3)v + (z_2 - z_3)w\right] \\ D_0 &= k\left[x_3u + y_3v + z_3w\right] \end{aligned} \tag{2.436}$$

such that they form a linear phase function based upon the coordinates of the triangle. Here *u,v,w* are the direction cosines given by *g*+*h* from (2.346) and (2.385).

The integral (2.432) can now be rewritten as [93]:

$$I_c = 2Ae^{jD_0}\left\{e^{jD_p}\left[\frac{C_0}{D_p\left(D_q - D_p\right)}\right] - e^{jD_q}\left[\frac{C_0}{D_q\left(D_q - D_p\right)}\right] - \frac{C_0}{D_qD_p}\right\} \tag{2.437}$$

This equation has numerical problems when one of the denominators nears zero and so the following special cases are defined using a Taylor series expansion:

- Case 1 $|D_p| < L_t$ and $|D_q \geqslant L_t|$

$$I_c = \frac{2Ae^{jD_0}}{jD_q} \sum_{n=0}^{\infty} \frac{(jD_p)^n}{n!} \left\{ \frac{-C_0}{n+1} + e^{jD_q} C_0 G(n, -D_q) \right\} \tag{2.438}$$

- Case 2 $|D_p| < L_t$ and $|D_q| < L_t$

$$I_c = 2Ae^{jD_0} \sum_{n=0}^{\infty} \sum_{m=0}^{\infty} \frac{C_0 (jD_p)^n (jD_q)^m}{(m+n+2)!} \tag{2.439}$$

- Case 3: $|D_p| \geqslant Lt$ and $|D_q| < L_t$

$$I_c = 2Ae^{jD_0} e^{jD_p} \sum_{n=0}^{\infty} \frac{(jD_q)^n}{n!} \left(\frac{C_0}{n+1} \right) G(n+1, -D_p) \tag{2.440}$$

- Case 4: $|D_p| \geqslant L_t$, $|D_q| \geqslant L_t$, and $|D_q - D_p| < L_t$

$$I_c = \frac{2Ae^{jD_0}}{jD_q} \sum_{n=0}^{\infty} \frac{(jD_p - jD_q)^n}{n!} \left\{ -C_0 G(n, D_q) + \frac{e^{jD_q} C_0}{n+1} \right\} \tag{2.441}$$

where Lt defines the limit of the Taylor Expansion and has been shown [93] to provide accurate results with value $L_t = 0.05$.

The function G is defined by:

$$G(n, \gamma) = \int_0^1 s^n e^{j\gamma s} ds \tag{2.442}$$

which can be evaluated recursively using the function:

$$G(n, \gamma) = \frac{e^{j\gamma} - nG(n-1, \gamma)}{j\gamma}, n \geqslant 1 \tag{2.443}$$

and setting the start up value to be :

$$G(0, \gamma) = \frac{e^{j\gamma} - 1}{j\gamma} \tag{2.444}$$

As a point of note, GPGPUs do not handle recursion very well and so a technique

called 'loop unrolling' has been used in this work to implement these equations on a graphical processing unit. This involves having specific expressions for each iteration of the recursive function in (2.443). It has been found however that this function almost always converges within 4 iterations and so the loop-unrolled code is not that cumbersome.

The triangular facet solution is now developed by recognising that the scattered electric field at the receiver is composed of both parallel and perpendicular polarisations:

$$\vec{E}_s(r,\theta,\phi) = E_\theta \hat{\vec{E}}_\theta + E_\phi \hat{\vec{E}}_\phi \tag{2.445}$$

where E_θ and E_ϕ are taken from (2.132) and (2.133) :

$$\begin{aligned} E_\theta &\simeq -jk\frac{e^{-jkr}}{4\pi r}\left(L_\phi + Z_0 N_\theta\right) \\ E_\phi &\simeq +\frac{jke^{-jkr}}{4\pi r}\left(L_\theta - Z_0 N_\phi\right) \end{aligned} \tag{2.446}$$

to provide

$$\vec{E}_s(r,\theta,\phi) = -jk\frac{e^{-jkr}}{4\pi r}\left(L_\phi \hat{\vec{E}}_\theta + Z_0 N_\theta \hat{\vec{E}}_\theta + Z_0 N_\phi \hat{\vec{E}}_\phi - L_\theta \hat{\vec{E}}_\phi\right) \tag{2.447}$$

$$\vec{E}_s(r,\theta,\phi) = \vec{E}_s^e + \vec{E}_s^m \tag{2.448}$$

which provides the scattered E field at the receiver in terms of the electric and magnetic sources of the surface current. Collecting terms related only to the electric field sources provides:

$$\vec{E}_s^e = -jk\frac{e^{-jkr}}{4\pi r} Z_0 \left(N_\theta \hat{\vec{E}}_\theta + N_\phi \hat{\vec{E}}_\phi\right) = -jk\frac{e^{-jkr}}{4\pi r} Z_0 \vec{N} \tag{2.449}$$

and collecting magnetic current sources provides:

$$\vec{E}_s^m = -jk\frac{e^{-jkr}}{4\pi r}\left(L_\phi \hat{\vec{E}}_\theta - L_\theta \hat{\vec{E}}_\phi\right) \tag{2.450}$$

This last equation contains cross terms that are difficult to remove. The nature of them however is interesting as these are the electric field components caused by the

magnetic surface currents, which are themselves coupled to incident electric fields.

The same analysis can be performed on the scattered magnetic fields:

$$H_s(r,\theta,\phi) = H_\theta \vec{\hat{H}}_\theta + H_\phi \vec{\hat{H}}_\phi \tag{2.451}$$

$$\vec{H}_s^m = -\frac{jke^{-jkr}}{4\pi r}\left(\frac{L_\theta}{Z_0}\vec{\hat{H}}_\theta + \frac{L_\phi}{Z_0}\vec{\hat{H}}_\phi\right) = -\frac{jke^{-jkr}}{4\pi r}\frac{1}{Z_0}\vec{L} \tag{2.452}$$

$$\vec{H}_s^e = \frac{jke^{-jkr}}{4\pi r}\left(N_\phi \vec{\hat{H}}_\theta - N_\theta \vec{\hat{H}}_\phi\right) \tag{2.453}$$

which again shows how the parallel and perpendicular electric and magnetic fields are intrinsically coupled. Fortunately, as discussed previously, if the facet is represented as a combination sheet then as long as the facet is completely planar (see (2.214)) then the magnetic and electric field components can be calculated separately and results combined to provide the total E field at the receiver. In this case the duality principle can be applied to determine the electric field caused by the magnetically conductive sheet:

$$\vec{E}_s^m = Z_0 \vec{H}_s^m = -\frac{jke^{-jkr}}{4\pi r}\vec{L} \tag{2.454}$$

and therefore:

$$\vec{E}_s(r,\theta,\phi) = \vec{E}_s^e + \vec{E}_s^m = -jk\frac{e^{-jkr}}{4\pi r}Z_0\left(\vec{N} + \vec{L}\right) \tag{2.455}$$

Equations (2.137) and (2.138) which describe the far field scattering components, N, L as :

$$N = \iint_s \vec{J}_s e^{jkh} ds' = \iint_s \left(\hat{a}_x J_x + \hat{a}_y J_y + \hat{a}_z J_z\right) e^{jkh} ds' \tag{2.456}$$

and

$$L = \iint_s \vec{M}_s e^{jkh} ds' = \iint_s \left(\hat{a}_x M_x + \hat{a}_y M_y + \hat{a}_z M_z\right) e^{jkh} ds' \tag{2.457}$$

with

$$h = r' \cos\psi_s \tag{2.458}$$

can now be used. The surface currents inside the integrand are still problematic. However if an assumption is made that the facet is small enough that there is no change in the facet material's composition over its surface then the resistivity

component *Rs* and conductivity component *Rm* are constant over the facet. It is therefore possible to remove them from within the surface integral to provide:

$$\vec{N} = \left(\hat{a}_x J_x + \hat{a}_y J_y\right) \iint_s e^{jkh} ds' \tag{2.459}$$

$$\vec{L} = \left(\hat{a}_x M_x + \hat{a}_y M_y\right) \iint_s e^{jkh} ds' \tag{2.460}$$

and so the complete solution for the scattered electric fields at a receiver from a planar triangular facet of realistic material composition (represented by a combination sheet) is:

$$\vec{E}_s(r,\theta,\phi) = \vec{E}_s^e + \vec{E}_s^m = -jk\frac{e^{-jkr}e^{-jkr_i}}{4\pi r} Z_0\left(\vec{J}' + \vec{M}'\right) I_c \tag{2.461}$$

with

$$I_c = \iint_s e^{jk(g+h)} ds' \tag{2.462}$$

which can be calculated using the technique on Ludwig Integration (2.437). Also

$$\vec{J}' = \left(J'_x \hat{x}'' + J'_y \hat{y}''\right) \tag{2.463}$$

$$\begin{aligned} J'_x &= 2\left(\frac{\cos\phi_i E_\theta^i}{Z_0\cos\theta_i + 2R_s} - \frac{\sin\phi_i E_\phi^i}{Z_0 + 2R_s\cos\theta_i}\right)\cos\theta_i \\ J'_y &= 2\left(\frac{\sin\phi_i E_\theta^i}{Z_0\cos\theta_i + 2R_s} + \frac{\cos\phi_i E_\phi^i}{Z_0 + 2R_s\cos\theta_i}\right)\cos\theta_i \end{aligned} \tag{2.464}$$

$$\vec{M}' = \left(M'_x \hat{x}'' + M'_y \hat{y}''\right) \tag{2.465}$$

$$\begin{aligned} M'_x &= 2Z_0\left(\frac{\cos^2\theta_i\cos\phi_i H_\theta^i}{1 + 2Z_0 R_m\cos\theta_i} - \frac{\sin\phi_i H_\phi^i}{\cos\theta_i + 2Z_0 R_m}\right) \\ M'_y &= 2Z_0\left(-\frac{\cos^2\theta_i\sin\phi_i H_\theta^i}{1 + 2Z_0 R_m\cos\theta_i} + \frac{\cos\phi_i H_\phi^i}{\cos\theta_i + 2Z_0 R_m}\right) \end{aligned} \tag{2.466}$$

The calculation of the RCS for a triangular facet can therefore be performed through the following steps:

1. Calculate the surface integral term using the incidence and scattering direction cosines for the triangle. (And adjusting the triangle's coordinates so that its

centre is at the origin)

2. Calculate $\cos\theta_i, \sin\phi_i, \cos\phi_i$

3. Calculate the surface coordinate components J'_x, J'_y, M'_x, M'_y

4. Translate the local current vectors $\vec{J''}, \vec{M''}$ into global vector coordinates $\vec{J}, \vec{M}$ and add them together

5. Take the vector dot product of the resulting vector with the vertical and horizontal sensor receive polarisations to get the V and H polarisations if required.

6. Convert the received E field into an RCS by comparing the E field to an isotropic scatterer at the facet's location.

The above steps have been performed on a CAD model representing a flat metal plate of area $1m^2$ composed of 36 smaller triangular facets. The results can be seen in Figure 2.18. The red line is the actual RCS measurements of the plate as a function of azimuth angle around the plate (the plate is illuminated from the position of $\theta = \pi$. It can be seen that it closely agrees with the theoretical RCS and sidelobe structure for a theoretical $(sin(x)/x)^2$ beam pattern from a flat plate. For comparison the RCS of the plate when modeled as a concrete panel and a water surface are also shown.

2.5 Summary

This chapter has derived the electromagnetic field strength at a receiver once scattered from a planar facet. It is worth noting that these equations are specified in terms of both the electrical resistivity of a material and its hypothetical magnetic conductivity thereby allowing the radar cross section from a wide range of materials, and metamaterials to be modelled.

In performing the derivation, it was noted that a fundamental difficulty exists in calculating the surface integral (in a computationally realistic time). Two solutions were provided: the first involved the use of rectangular scattering primitives and

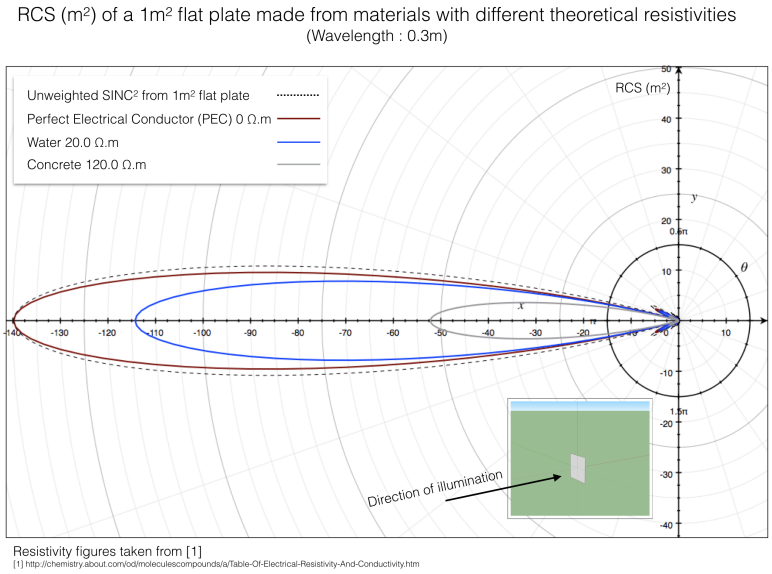


Figure 2.18: RCS of flat plate modeled using the triangular facets and physical optics

the second involved triangular scattering primitives. In both cases a simulation was developed and compared to theoretical results.

A major advantage of triangular facets is that they can approximate any three dimensional surface without introducing holes within the topology. This is not achievable with planar rectangular facets. For this reason the rest of this thesis will be specifically related to the planar triangle. The next chapter will consider the geometric calculations and considerations required to translate an arbitrary computer aided design (CAD) model into a planar triangular mesh of sufficient quality that can be ray-traced in a meaningful way in a radar simulator.

Chapter 3

All About Triangles

3.1 Introduction

A discussion on the nature of triangles may seem like an unusual chapter in a thesis on electromagnetic scattering. The subject is however a key component of any type of ray-tracing problem and one frequently left unconsidered in more radar-related texts. Fortunately the computer games industry and computer aided design (CAD) modelling community have expended a significant amount of effort in reducing the challenges associated with decomposing a CAD model into facet primitives. In this chapter, the key steps required to convert a CAD model into a usable data structure for EM scattering calculations will be considered.

A CAD model can be formed from a wide range of scattering primitives, planar and non-planar. As this thesis focuses on rapid EM field prediction using the physical optics approximation, planar facets will be the only type considered. The most useful planar facet to build a CAD model from is the triangle.

The chapter starts with a consideration of the deconstruction of a CAD model into triangular facets and the differences between a *triangle mesh* and a *triangle soup*.

The triangle mesh can be composed of many varied triangles which may cause errors or inconsistent results during ray tracing. This problem will be discussed together with a solution- the decomposition of the mesh into Delaunay triangles.

The resulting mesh is then abstract in terms of the electromagnetic properties of

the underlying CAD model. A technique to *paint* the mesh with electrically resistive (and magnetically conductive) properties is presented together with a method for applying surface roughness characteristics.

Finally the subject of ray-triangle intersection will be addressed and a technique, commonly used in the computer games industry, to minimise the number of floating point operations will be presented.

3.2 Mesh or Soup?

A major complaint levied at SAR simulations is the time it takes to produce a CAD model of the scene being simulated. With SARCASTIC, the assumption is made that a pretty good representation of a SAR scene can be made using a lower level of precision for the CAD models. Whilst this may be inaccurate for precise detail, it is a useful approximation for such things as general scene understanding and multipath effects recognising that most SAR sensors do not produce imagery at the level of detail of the precise CAD models used. Having made this assumption, it is possible to more readily raid the internet hobbyist community CAD models for more rapid scene understanding. In this work we used two applications for CAD model design: Sketchup® [96] and Autodesk Maya® [97]. In particular, Sketchup® has an extensive warehouse of of models [98] covering a wide range of buildings and equipment and has the advantage of being freely available.

Both of these applications allow a CAD model to be exported in an open source format called Collada™ [99] which can be readily transformed into a simple Polygon File Format, or .PLY format [100].

One of the reasons for converting from Collada™ to a more simple file format is to validate the integrity of the facets and ensure that they are well formed. Models can be constructed in a number of ways and exporting the models into a well ordered mesh is a fundamentally difficult process. Three common issues often arise:

- Producing individual triangular facets each with their own vertices creates 3 times as many vertices as there are triangles
- Triangles can be degenerate - i.e. have two common vertices or two vertices

very close together

- Producing a triangle soup rather than a triangle mesh. A triangle soup contains triangles that may be adjacent to each other but the vertices of adjacent triangles are not common. A triangle mesh strictly ensures that each triangle has common vertices with any of its adjacent triangles. The difference between these can be seen in figure 3.1

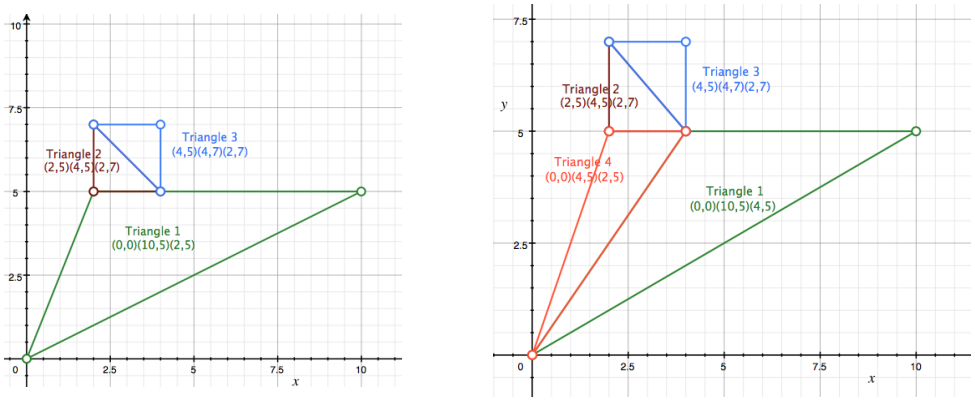


Figure 3.1: (Left) : A *Triangle Soup* contains adjacent triangles that do not share vertices. e.g. Triangle 1 and Triangle 2. (Right): A *Triangle Mesh* has all adjacent triangles sharing common vertices

The first two problems are related in that, in order to further process the dataset, the vertices and facets must be rationalised. This is the purpose of the COLLADA-TO-PLY file conversion. The algorithm proceeds as in Algorithm 1.

In this algorithm, the UNIQUE function (line 7) removes all duplicate vertices in the *vertexArray* ensuring that each point only occurs once in the output file. The SORT function (line 6) ensures that this can be performed quickly.

The third common issue is more complex and, although unintended, can appear in triangle facet output from common CAD packages. Because of this and to ensure robustness later on, it is critical to run a pass over the resulting output of Algorithm 1 to guarantee that the scene is a triangle mesh. The key to solving this problem comes from recognising that in a triangle mesh each side of every triangle can have exactly 1 or 2 triangles. In a triangle soup (such as Figure 3.1) a triangle edge can have more than 2 adjacent triangles. We call this type of triangle *polygamous*. The

Algorithm 1 Converting Collada™ triangle soup to a .PLY file Triangle Mesh

```
1:  procedure colladaToPlyFile(in, out)
2:      Triangle Soup M ← in                          ▷ Read Triangle Soup into M
3:      for Triangle T ∈ M do
4:          if not isDegenerate(T) then
5:              vertexArray ← T · vertices
6:      vertexArray ← sort(vertexArray)
7:      vertexArray ← unique(vertexArray)
8:      ▷ rebuild triangles using references to vertices
9:      for Triangle T ∈ M do
10:         t ← new triangle
11:         for v ← vertex V ∈ T do
12:             i ← index of v in vertexArray
13:             t.v ← i
14:             triangleArray ← t
15:     out ← vertexArray                             ▷ Write Vertices to out
16:     out ← triangleArray          ▷ Write Triangles (using vertex indices) to out

17: function isDegenerate(Triangle T)
18:     if T · Area < ε then                          ▷ ε is a small number
19:         return True
20:     else
21:         return False
```

algorithm to remove polygamous triangles from the soup and guarantee a mesh is called monogamise and can be seen in Algorithm 2.

The algorithm loops through every triangle in the soup and then checks that no vertex within the soup lies on any of the triangle's edges. If a vertex is found on an edge then the triangle is split into two, the original triangle is erased and the two new triangles added to the soup. The search is then reapeated until no triangle is polygamous.

This can be quite an intensive process due to the requirement to check each vertex against each triangle side. Additionally it cannot be run in parallel as each polygamous triangle found results in it being replaced by two new triangles that each have to be added to the triangle array for testing. Fortunately this operation only has to occur once per scene as a integrity-check function of the required CAD model. Additionally, unrolling the triangle sides (rather than iterating through them) and

Algorithm 2 MONOGAMISE - Convert a triangle soup to a triangle mesh

```
1:  procedure MONOGAMISE(TriangleMesh m)
2:      do
3:          triAdded = False
4:          for Triangle T ∈ m do
5:              for v ∈ vertexArray do
6:                  if v lies on side T.side[0] then
7:                      t1 ← NEW TRIANGLE( T·vertex[0] → v → T·vertex[2] )
8:                      t2 ← NEW TRIANGLE( T·vertex[2] → v → T·vertex[1] )
9:                  else if v lies on side T.side[1] then
10:                     t1 ← NEW TRIANGLE( T·vertex[1] → v → T·vertex[0] )
11:                     t2 ← NEW TRIANGLE( T·vertex[0] → v → T·vertex[2] )
12:                 else if v lies on side T.side[2] then
13:                     t1 ← NEW TRIANGLE( T·vertex[2] → v → T·vertex[1] )
14:                     t2 ← NEW TRIANGLE( T·vertex[1] → v → T·vertex[0] )
15:                 triangleArray·PUSHBACK(t1,t2)
16:                 triangleArray·ERASE(T)
17:                 triAdded ← True
18:                 break
19:             if triAdded then
20:                 break
21:     while triAdded = True
```

using BREAK to quickly exit loops when a polygamous triangle is found, reduces the computational burden.

3.3 Delaunay Triangulation

Sometimes the output CAD model can consist of triangles that have a very small internal angle - called *skinny triangles*. Whilst being perfectly valid and, in some cases necessary for the CAD model they can cause problems in the calculation of the surface integral of the incident EM field. To correct for this, an additional pre-processing step is required to maximise a triangle's vertex angles. This is called *Delaunay triangulation*.

Delaunay Triangulation is named after Boris Delaunay [101] who developed the technique in 1934 as a way to address the problem of "skinny triangles". The general idea is to introduce points into a triangular mesh such that each of the vertices for any triangle lie on the edge of a circle and there are no other points within the

circle. This means that the triangles making up any quadrilateral have the sum of their angles in the vertex away from the adjoining line being less than π radians (Figure 3.2).

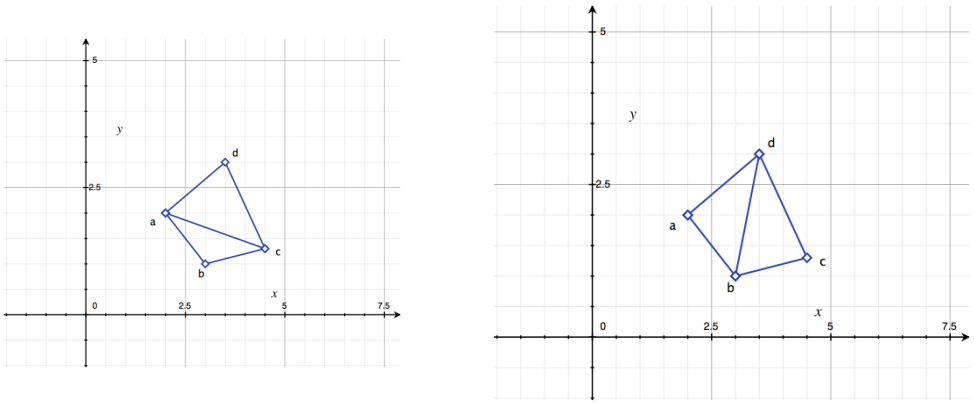


Figure 3.2: Delaunay Triangulation. (Left) : Triangle mesh $abcd$ does not meet the Delaunay condition as the sum of the angles at vertex $\angle abc$ and vertex $\angle cda$ is greater than π. (Right) : The same mesh can be Delaunay triangulated by forming triangles $\triangle abd$ and $\triangle bcd$ such that $\angle abd + \angle bcd < \pi$

Fortunately the Delaunay triangulation process only has to occur once during scene pre-processing and there are many excellent open-source libraries available that perform the task rapidly [102], [103].

3.4 Material Properties

Rather than assuming that all materials in a CAD model are PEC surfaces, SARCASTIC attempts to simulate the electromagnetic response of a facet based upon its material composition. This is performed in two ways:

- The calculations for the scattered electromagnetic fields take into account the electrical conductivity and magnetic permeability of a range of different materials (Chapter 2).

- The surface roughness of a range of materials is simulated during Delaunay triangulation.

In both cases, the CAD model is *painted* during its construction with different material properties. These are stored within the Collada™file and preserved during

Material	Correlation Length (metres)	Roughness (metres)	Resistivity ($\Omega \cdot m$ at $20°C$)
PEC	0.05	0.0	0.0
Asphalt	0.5	0.05	1.0×10^{18}
Brick	0.1	0.001	1.0×10^{18}
Concrete	0.2	0.01	120.0
Metal	0.6	0.0	1.0×10^{-8}
Roofing	0.1	0.1	1.0×10^{18}
Vegetation	0.1	0.1	2000.0
Water	0.1	0.005	2.0×10^{1}
Wood	0.1	0.001	1.0×10^{14}
Grass	0.01	0.04	2000.0

Table 3.1: Default material properties used in SARCASTIC. Electrical properties are taken from [104].

the conversion to Polygon File Format. An example of the default material properties used in SARCASTIC can be seen in Table 3.1 (which can be changed by the user at run time).

Surface roughening is presented here as a way to simulate the random nature of a none-specular surface. It recognises that whilst the scattered electromagnetic field of a single facet can be calculated from its electrical properties, a significant amount of the collected reflected energy is dominated by the shape and interaction of a number of facets close to each other. The work builds on previous research by applying a distortion to a surface equivalent to the root mean square surface roughness [105]. First the mean correlation length is applied during the Delaunay triangulation process. This is then followed by a distortion pass to adjust the new mesh vertices away from the original surface by a mean surface roughness.

In implementing this process there are several subtleties that must be accounted for. First, only triangles that have the same surface normal, the same material properties and form a contiguous region can be considered in a single Delaunay triangulation surface roughening step. Additionally it is critically important that the perimeter of the new region remains undistorted. This is to ensure that it remains contiguous with surrounding regions in order to prevent holes and gaps being created between adjoining triangles in the two regions.

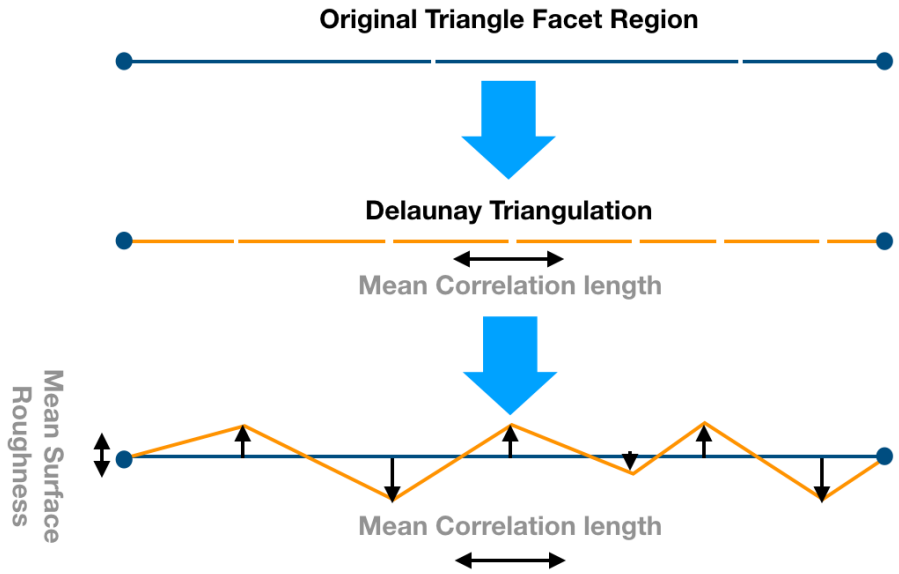


Figure 3.3: Creating a Rough Surface Mesh. Contiguous planar facets (Blue lines, top) are selected for processing. The mesh is recursively Delaunay triangulated until a mean correlation length is achieved (middle). Finally each interior vertex point is distorted by a random amount according to the mean surface roughness (bottom)

Algorithm 3 shows the steps required to pre-process the CAD model into a triangular mesh ready for processing by SARCASTIC.

Algorithm 3 Materialise - Perform Delaunay triangulation and roughen surface according to CAD material properties

```
1: procedure MATERIALISE(TriangleMesh m)
2:     while m not empty do
3:         commonMesh = GROWTRIANGLES(m)
4:         maxSize, roughness ← commonMesh[0].MATERIAL        ▷ all the same
5:         commonMesh ← DELAUNAYTRIANGULATE(commonMesh, maxSize)
6:         commonMesh ← ROUGHEN(commonMesh, roughness)
7:         for Triangle t ∈ commonMesh do
8:             newMesh.PUSH_BACK(t)
9:     return newMesh
```

A key function of the algorithm is the GROWTRIANGLES function. This function takes the last triangle in the input triangle mesh and uses it to determine all other triangles within the mesh that it shares a region with. The triangles in the new region are removed from the original mesh and returned as a *commonMesh*. The algorithm proceeds in this way until the input mesh has been emptied and each region has been

determined, Delaunay triangulated, roughened and added to the output *newMesh*.

The results from the MATERIALISE algorithm can be seen in Figure 3.4.

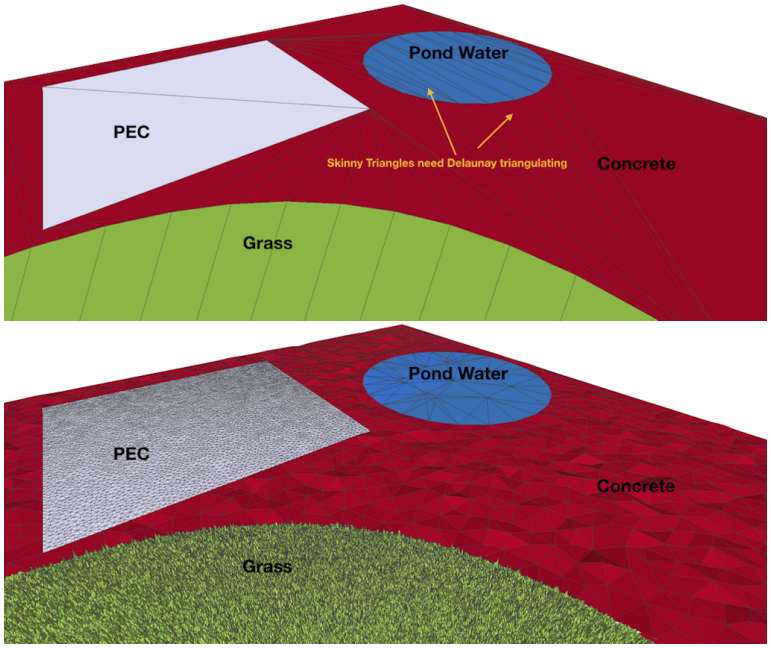


Figure 3.4: Application of MATERIALISE Algorithm. (Top) : A simple CAD model that has been *painted* with different colours depicting different material types. These are passed through the MATERIALISE algorithm (Bottom). The algorithm first delimits the common regions within the model and performs a Delaunay Triangulation on them making sure that each material's triangles are constrained to a maximum *correlation length*. A surface roughness filter is then applied to simulate the scattering that might occur from the surface

3.5 Ray-Triangle Intersection Optimisation

So far this chapter has focused on the pre-processing steps required to take a CAD model and reconstruct it in such a way as to be useful and efficient for SAR ray-tracing. There is one additional step involved with the scene pre-processing that has a significant impact on the performance of a radar ray tracer : Ray-Triangle Intersection Optimisation. This section will now describe a way to significantly improve ray-tracing efficiency.

The calculation of the intersection between a ray and a scene primitive is of paramount importance for scene rendering. There has been considerable work in this area detailing different approaches to reduce calculation time. Each approach has different properties such as the number of floating-point calculations, memory

requirements or numerical accuracy. In addition, different problem sets, implementations and hardware platforms make it difficult to compare one technique against another and as a result many debates have engulfed the ray-tracing community.

As the approach taken here is to consider only triangle primitives and as the focus is for a general compute engine (such as Intel CPU cores and GPU cards) rather than specialist computer systems, it is comparatively easier to select a suitable approach. For completeness, the main works in the area of optimised ray-triangle intersection calculations are [106–109]. All of these techniques provide subtle variations in performance often trading specialised hardware and complexity for speed. The preferred solution used in this document however is the technique developed for the RTRT (real time ray tracing) / OpenRT (Open Ray Tracing) solution as they are firstly well documented in [110] and they are ideally optimised for Intel streaming SIMD extensions (SSE). This means that with only a little effort they can be optimised for a more generic high performance environment such as *OpenCL* / *CUDA* and GPU compute platforms.

This section will start with the generic formulation of a ray-triangle intersection calculation. It will then proceed to develop an optimised approach with a considerably reduced computation instruction count.

3.5.1 Generic Formulation of Ray-Triangle Intersection

Given a ray R defined by its origin O and direction D, the location of any point along the ray is given by 3.1 as:

$$R(t) = O + D(t) \tag{3.1}$$

Equally, a triangle is defined by the coordinates of its vertices A, B, C and so the task of ray-triangle intersection of one of solving for the location H where:

$$H = R(t_{hit}) \tag{3.2}$$

Consider Figure 3.5. In this figure the triangle is represented by the plane AB and the blue line represents the ray $R(t)$. The distance t_{plane} from the origin of the

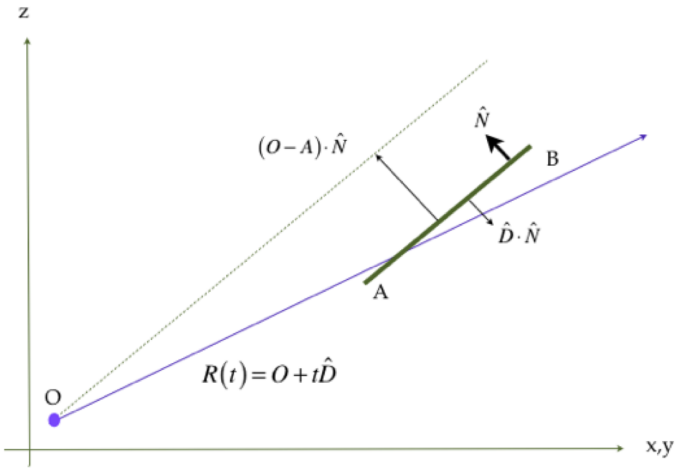


Figure 3.5: Derivation of a Ray Intersecting a Plane

ray to the point of intersection of the ray with the plane of the triangle is therefore given by:

$$t_{plane} = -\frac{(O-A)\cdot N}{D\cdot N} \tag{3.3}$$

Once t_{plane} is found it is then possible to determine if it lies within acceptable bounds of t such that $t_{min} < t < t_{max}$. The normal N can be pre-calculated when the scene is prepared through :

$$N = (B-A)\times(C-A) \tag{3.4}$$

Once the distance test is passed then the coordinate of the ray intersecting with the plane of the triangle is provided by:

$$H = R\left(t_{plane}\right) = O + D\left(t_{plane}\right) \tag{3.5}$$

The next test is to determine if the intersection point H lies within the boundary of the triangle or outside it. This is considerably more difficult and is therefore the main source of computation optimisation. A brute force solution to this problem revolves around calculating the angle subtended at the intersection point between each of the vertices in the triangle. The calculation proceeds as in Algorithm 4.

Although Algorithm 4 is relatively simple, it consists of many potentially

Algorithm 4 Determine whether a hit point H is inside triangle $\triangle ABC$

```
1: procedure INSIDETRIANGLE(H, △ABC)
2:     AB ← ∠AHB
3:     BC ← ∠BHC
4:     CA ← ∠CHA
5:     if (AB + BC + CA) ≤ 2π then
6:         return True                    ▷ Point is inside triangle
7:     else
8:         return False                   ▷ Point is outside triangle
```

Operation	Add	Subtract	Multiply	Divide	Sqrt	Trig
$V_1 = B - H$		3				
$V_2 = H - A$		3				
$V_1 \cdot V_2$	2		3			
$\lvert V_1 \rvert$	2		3		1	
$\lvert V_2 \rvert$	2		3		1	
$\lvert V_1 \rvert \lvert V_2 \rvert$			1			
$V_1 \cdot V_2 / \lvert V_1 \rvert \lvert V_2 \rvert$				1		
arccos						1
Total Operations	6	6	10	1	2	1
Instruction Latency (cycles)	3	3	5	27	27	100
Total Clock Cycles/Instruction	18	118	50	27	54	100
Total Cycles for Calculation			**267**			

Table 3.2: Nominal CPU clock cycles to determine whether a point is inside a triangle using [111]

expensive individual operations:

$$\angle AHB = \arccos\left(\frac{V_1 \cdot V_2}{|V_1||V_2|}\right) \tag{3.6}$$

where

$$V_1 = B - H \tag{3.7}$$

and

$$V_2 = H - A \tag{3.8}$$

The number of CPU clock cycles required to identify whether a point is within a triangle is summarised in Table 3.2. The clock cycles required for each floating point operation is taken from [111] and assumes the Intel Core i7 Nehalem processor.

This simple algorithm is therefore quite computationally expensive, especially when tests have to be performed with many rays and many triangles for each ray. Efficiencies can be found however [107,112] by considering a coordinate conversion into a Barycentric Coordinate system.

3.5.2 Barycentric Coordinates

Barycentric coordinates were first proposed by August Ferdinand Mœbius in 1827 as an extension to Ceva's Theorum which states that (Figure 3.6) :

> In a triangle $\triangle ABC$, three lines AD, BE and CF intersect at a single point K if, and only if:
>
> $$\frac{AF}{FB}\frac{BD}{DC}\frac{CE}{EA} = 1$$

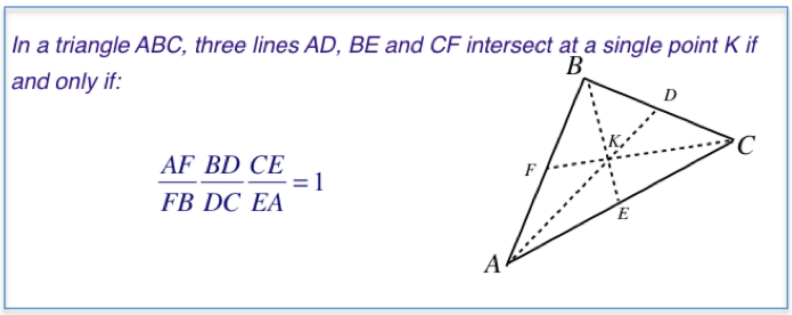


Figure 3.6: Ceva's Theorum

It is intuitive from Figure 3.6 that, for any point K within $\triangle ABC$, there exists three masses, m_a, m_b, m_c such that, if placed at the corresponding vertices of the triangle, their centre of gravity (barycentre) coincides with the point K. By extension, if A, B and C are already known then the application of weights, α, β and γ to each of these masses would produce the same result.

Mœbius defined the weights α, β, γ as the barycentric coordinates of K for the coordinate system defined by A, B, C. As defined, the barycentric coordinates are not unique. Masses $k\alpha$, $k\beta$, and $k\gamma$ have exactly the same barycentre for any value of $k > 0$. Thus barycentric coordinates are a form of general homogeneous coordinates that are used in many branches of mathematics. With one additional condition:

$$\alpha + \beta + \gamma = 1 \tag{3.9}$$

The barycentric coordinates are defined uniquely for every point inside the triangle. (Barycentric coordinates that satisfy 3.9 are known as areal coordinates because, assuming the area of $\triangle ABC$ is 1, the weights $k\alpha$, $k\beta$, and $k\gamma$ are equal to the areas of triangles $\triangle KBC$, $\triangle KAC$, and $\triangle KAB$ respectively.) Since the centre of gravity of any two points lies on the connecting segment, $\alpha = 0$ for points on BC, $\beta = 0$ for points on AC, and $\gamma = 0$ for points on AB. Vertices A, B, C have coordinates $(1,0,0)$, $(0,1,0)$, and $(0,0,1)$, respectively. As a result the sum on $k\alpha$, $k\beta$, and $k\gamma$ must be one.

3.5.3 Simplification Through Projection

Using triangle $\triangle ABC$ to define the barycentric coordinate system, the barycentric coordinate of H can be written as:

$$H = \alpha A + \beta B + \gamma C \tag{3.10}$$

Here we recognise that α, β and γ are the areal coordinates of the system and subject to usual barycentric coordinate constraints $\alpha + \beta + \gamma = 1$. Therefore, if H is inside the triangle, α, β and γ must all be positive. Once these coefficients have been found it is possible to determine if H is inside the triangle by checking whether the conditions

$$0 \leq \alpha \leq 1, \quad 0 \leq \beta \leq 1, \quad 0 \leq \gamma \leq 1 \tag{3.11}$$

are fulfilled. This check can be made more efficient through careful implementation (Algorithm 5) which fails at the earliest opportunity therefore reducing unnecessary calculations.

Algorithm 5 Test whether barycentric hit point $H(\alpha, \beta, \gamma)$ is inside triangle.

```
1: procedure INSIDETRIANGLE(α, β, γ)
2:     if β < 0 then return False
3:     if γ < 0 then return False
4:     if β + γ > 1 then return False
5:     else return True
```

An interesting feature of barycentric coordinates is that the coordinates of the hit point H do not change if the triangle is projected into any other plane (that is

not orthogonal to the plane *ABC*), (Figure 3.7). This means that the calculations

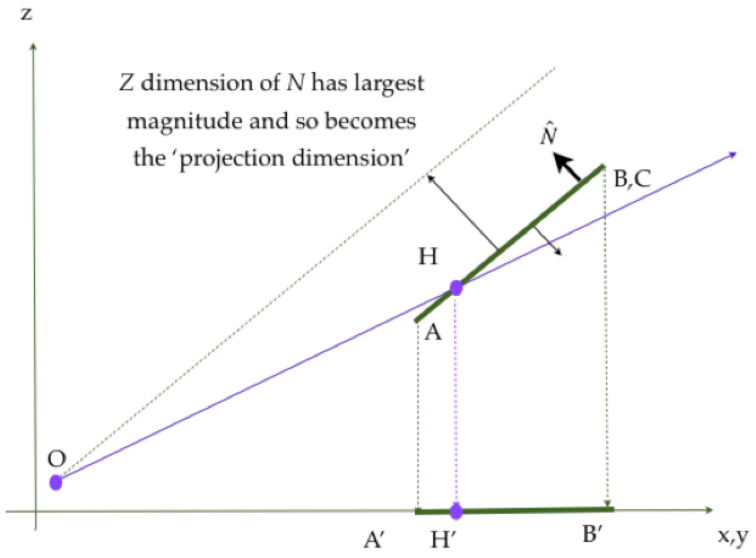


Figure 3.7: Barycentric coordinates are preserved after projection and so the triangle is projected onto the Cartesian axis with the largest component of the triangle's normal

can be simplified by projecting the coordinates into a 2 dimensional plane. To maintain calculation stability the plane should be selected that maximises the area of the triangle. This is the same dimension as the dimension of normal N that has the largest absolute value. After projection the calculations can be performed more efficiently in two dimensions ((Figure 3.7), where projection into the XY plane yields:

$$H' = \alpha A' + \beta B' + \gamma C' \tag{3.12}$$

where A', B', C' and H' represent the projected points of A, B, C and H respectively. Substituting $\alpha = 1 - \beta - \gamma$ into 3.12 and rearranging yields:

$$H' = (1 - \beta - \gamma) A' + \beta B' + \gamma C' \tag{3.13}$$

$$H' = A' - \beta A' - \gamma A' + \beta B' + \gamma C' \tag{3.14}$$

$$H' - A' = \beta (B' - A') + \gamma (C' - A') \tag{3.15}$$

Assigning the following for convenience:

$$b = C' - A' \tag{3.16}$$

$$c = B' - A' \tag{3.17}$$

$$h = H' - A' \tag{3.18}$$

and so 3.15 can be represented as:

$$\begin{pmatrix} c_u & b_u \\ c_v & b_v \end{pmatrix} \begin{pmatrix} \beta \\ \gamma \end{pmatrix} = \begin{pmatrix} h_u \\ h_v \end{pmatrix} \tag{3.19}$$

$$\begin{pmatrix} \beta \\ \gamma \end{pmatrix} = \begin{pmatrix} h_u \\ h_v \end{pmatrix} \frac{1}{\det|cb|} \begin{pmatrix} b_v & -b_u \\ -c_v & c_u \end{pmatrix} \tag{3.20}$$

where subscripts u and v denote the two Cartesian coordinate axes that the triangle has been projected into (x and y in Figure 3.7). This provides

$$\beta = \frac{b_v h_u - b_u h_v}{\det|cb|} \tag{3.21}$$

$$\gamma = \frac{c_u h_v - c_v h_u}{\det|cb|} \tag{3.22}$$

or:

$$\beta = \frac{\det|bh|}{\det|bc|}, \ \gamma = \frac{\det|hc|}{\det|bc|} \tag{3.23}$$

In two dimensions this can be expressed efficiently as:

$$\beta = \frac{b_u h_v - b_v h_u}{b_u c_v - b_v c_u}, \ \gamma = \frac{h_u c_v - h_v c_u}{b_u c_v - b_v c_u} \tag{3.24}$$

and so all that is required to determine whether the point H denoted by (h_u, h_v) is inside the triangle is to calculate the parameters b_u, b_v, c_u, c_v and substitute them into equations 3.24. This is a desirable solution as each of the values for b and c can be calculated when the scene is pre-processed. As a result, an essential step

Operation	ADD	SUBTRACT	MULTIPLY	DIVIDE	COMPARE
Calculate β		2	4	1	
Calculate γ		2	4	1	
if $\beta < 0$					1
if $\gamma < 0$					1
if $(\beta + \gamma) > 1$	1				1
Instruction (best case)	0	2	4	1	1
Instructions (worst case)	1	4	8	2	3
Latency (clock cycles)	3	3	5	27	1
Cycles / instruction (best)	0	6	20	27	1
Cycles / instruction (worst)	3	12	40	54	3
Total cycles to test if H inside $\triangle ABC$		**Best**	54	**Worst**	112

Table 3.3: CPU clock cycles required to determine whether a hit point is within a triangle when barycentric coordinates are used

for the scene builder is to first calculate and store the barycentric parameters for each triangle. When ray tracing is required it is these values that are checked for intersection. For comparison with Table 3.2, Table 3.3 is provided to show the number of clock cycles required to determine whether a hit point is within a triangle using barycentric coordinates.

It can be seen from this that in the worst-case situation, where all three barycentric parameters have to be tested, only 112 CPU clock cycles are performed. This compares favorably with the 267 CPU clock cycles required if no coordinate transformation is performed (3x speed up).

3.5.4 Implementation Efficiencies

In addition to performance improvements through mathematical techniques, efficient computer programming can also improve performance. Once the projection dimension (K) has been determined by inspection of the triangle's normal, the projection coordinates u and v are usually calculated using Algorithm 6. In this algorithm MOD is the standard modulo operator. The modulo operator is quite expensive in terms of central processing unit (CPU) cycles requiring two move operations and a divide operation for a total of 28 clock cycles (in the 'C' language). A much simpler solution is to use the value of k to reference a lookup table as in Algorithm 7. Additional

Algorithm 6 Determining the projection plane for a triangle

```
1: ▷ k is index of largest axis in normal N (comprised of x,y,z)
2: procedure projectionPlane(k)
3:     u ← mod(k+1,3)
4:     v ← mod(k+2,3)
5:     return u,v
```

Algorithm 7 Determining the projection plane for a triangle using a lookup table

```
1: ▷ k is index of largest axis in normal N (comprised of x,y,z)
2: procedure projectionPlaneLUT(k)
3:     modulo[5]=[0,1,2,0,1]
4:     u ← modulo[k+1]
5:     v ← modulo[k+2]
6:     return u,v
```

savings can be found through closer examination of the calculations involving the normal and recognising that if $K = N_k$ is the projection dimension (i.e. the largest coordinate value of the normal) then $N \cdot K$ can never be zero. It is therefore possible to divide the normal N by $N \cdot K$ to provide:

$$N' = \frac{N}{N \cdot K} \tag{3.25}$$

From 3.3 t can be rewritten as:

$$t = -\frac{(O - A) \cdot N'}{D \cdot N'} = \frac{A \cdot N' - (O \cdot N')}{D \cdot N'} \tag{3.26}$$

and so t becomes:

$$t = \frac{d - (O_u N'_u + O_v N'_v + O_k N'_k)}{D_u N'_u + D_v N'_v + D_k N'_k} \tag{3.27}$$

with:

$$d = A \cdot N', \quad N'_u = \frac{N_u}{N_k}, \quad N'_v = \frac{N_v}{N_k}, \quad N'_k = \frac{N_k}{N_k} = 1 \tag{3.28}$$

and so recognising $N'_k = 1$ removes two additional multiplications from the equation for t when calculating the ray intersection and so t becomes:

$$t = \frac{d - (O_u N'_u + O_v N'_v + O_k)}{D_u N'_u + D_v N'_v + D_k} \tag{3.29}$$

Operation	Add	Subtract	Multiply	Divide	Compare
Calculate β	3		2		
Calculate γ	3		2		
if $\beta < 0$					1
if $\gamma < 0$					1
if $(\beta + \gamma) > 1$	1				1
Instruction (best case)	3	0	2	0	1
Instructions (worst case)	7	0	4	0	3
Latency (clock cycles)	3	3	5	27	1
Cycles / instruction (best)	9	0	10	0	1
Cycles / instruction (worst)	21	0	20	0	3
Total cycles to test if H inside $\triangle ABC$		**Best**	20	**Worst**	44

Table 3.4: CPU clock cycles required to determine whether a hit point is within a triangle when barycentric coordinates are used and by pre-factoring the triangle parameters

Looking again at the projection of barycentric coordinates (3.24) and expanding it using 3.16, 3.17 and 3.18, it can be rewritten as:

$$\beta = \frac{b_u(H' - A')_v - b_v(H' - A')_u}{b_u c_v - b_v c_u} \tag{3.30}$$

or:

$$\beta = \frac{b_u H_v - b_u A_v - b_v H_u + b_v A_u}{b_u c_v - b_v c_u} \tag{3.31}$$

$$\beta = \frac{b_u H_v - b_v H_u}{b_u c_v - b_v c_u} + \frac{b_v A_u - b_u A_v}{b_u c_v - b_v c_u} \tag{3.32}$$

$$\beta = \frac{b_u}{b_u c_v - b_v c_u} H_v + \frac{-b_v}{b_u c_v - b_v c_u} H_u + \frac{b_v A_u - b_u A_v}{b_u c_v - b_v c_u} \tag{3.33}$$

$$\beta = \kappa_{bu} H_v + \kappa_{bv} H_u + \kappa_{bd} \tag{3.34}$$

which provides an efficient way to pre-calculate and store the κ constants during scene pre-processing and allows β to be calculated during the ray tracing run with only two multiplies and three additions (nineteen clock cycles in total) (Table 3.4).

The same analysis can now be applied for the γ term (provided here for com-

pleteness). From 3.24, and 3.16, 3.17 and 3.18:

$$\gamma = \frac{(H' - A')_x c_y - (H' - A')_y c_x}{b_x c_y - b_y c_x} \tag{3.35}$$

$$\gamma = \frac{c_v H_u - c_v A_u - c_u H_v + c_u A_v}{b_u c_v - b_v c_u} \tag{3.36}$$

$$\gamma = \frac{c_v H_u - c_v A_u - c_u H_v + c_u A_v}{b_u c_v - b_v c_u} \tag{3.37}$$

$$\gamma = \frac{c_v}{b_u c_v - b_v c_u} H_u - \frac{c_u}{b_u c_v - b_v c_u} H_v + \frac{c_u A_v - c_v A_u}{b_u c_v - b_v c_u} \tag{3.38}$$

$$\gamma = \kappa_{cu} H_v + \kappa_{cv} H_u + \kappa_{cd} \tag{3.39}$$

In summary the ray-triangle intersection process can be condensed into the steps in Algorithm 8

Algorithm 8 Efficient ray-triangle intersection calculation

```
1:  procedure preBuildTriangles(Mesh m)   ▷ Run once during pre-processing
2:      modulo[5]=[0,1,2,0,1]
3:      for triangle t ∈ m do
4:          k ← index of largest element (x,y,z) of N
5:          u ← modulo[k+1]
6:          v ← modulo[k+2]
7:          Calculate N'_u, N'_v and d and store
8:          Calculate κ_bu, κbv and κ_bd and store
9:          Calculate κ_cu, κcv and κ_cd and store
10:     return

    // acc are the accelerated triangle parameters calculated previously
11: procedure intersect(acc, Ray r)
12:     Determine t for ray intersecting with triangle plane (3.29)
13:     Calculate β (3.34)
14:     if β < 0 then return missed
15:     Calculate γ (3.39)
16:     if γ < 0 then return missed
17:     if (β + γ) > 1 then return missed
18:     return u, v                ▷ hit at location u, v on projected triangle plane
```

When implemented this algorithm can determine a ray-triangle intersection in as few as 20 CPU clock cycles - a considerable improvement over the original 267 clock cycles. Of course this improvement is bought at a cost. The scene has

to be pre-processed in order to determine the correct barycentric factors. This is not unreasonable when one considers that even with a dynamic scene the number of times it has to be reprocessed is small compared to the number of rays being rendered.

The algorithm detailed here represents the current best-practice for ray-triangle intersection calculations and is used throughout the gaming and movie industry. Performance can differ greatly however depending on how the algorithm is implemented. For this reason the source code is provided in Listing 3.1

Listing 3.1: C code to perform efficient ray-triangle intersection on a GPGPU

```
typedef struct {
    REAL cell[3];
} VectorH ;
typedef struct Ray {
    VectorH org;     // Origin
    VectorH dir;     // Direction
} Ray;
typedef struct Hit {
    REAL dist;      // Distance t of hit
    int trinum;      // Id of triangle that was hit
    REAL u;          // barycentric beta
    REAL v;          // barycentric gamma
} Hit;
typedef struct Triangle {
    int  triNum;     // Triangle ID
    REAL d;          // Constant of plane equation
    REAL nd_u;       // Normal.u / normal.k
    REAL nd_v;       // normal.v / normal.k
    int k;           // projection dimension
    REAL kbu;
    REAL kbv;
    REAL kbd;
    REAL kcu;
    REAL kcv;
    REAL kcd;
} Triangle;

void triangleIntersect(__global Triangle *tri, Ray *ray, Hit *hit)
{
    const unsigned int modulo[5] = {0,1,2,0,1};
    int ku = modulo[tri->k+1];
    int kv = modulo[tri->k+2];

    // Division is expensive so do this early
    const double nd = 1.0/(ray->dir.cell[tri->k] +
        ((tri->nd_u) * ray->dir.cell[ ku ]) + ((tri->nd_v) * ray->dir.cell[ kv ]) );
    const double thit = (tri->d - ray->org.cell[tri->k] - tri->nd_u * ray->org.cell[ ku ]
         - tri->nd_v * ray->org.cell[ kv ]) * nd;

    // check for valid distance.
    if ( !(hit->dist > thit && thit >  EPSILON  ) ) return;

    // compute hitpoint positions on uv plane
    const double hu = (ray->org.cell[ku] + thit * ray->dir.cell[ ku ]);
    const double hv = (ray->org.cell[kv] + thit * ray->dir.cell[ kv ]);

    // check first barycentric coordinate
    const double beta = (hu * tri->kbu + hv * tri->kbv + tri->kbd);
    if (beta < 0.0f) return ;

    // check second barycentric coordinate
    const double gamma = (hu * tri->kcu + hv * tri->kcv + tri->kcd);
    if (gamma < 0.0f) return;

    // check third barycentric coordinate
    if (beta+gamma > 1.0f) return ;

    // have a valid hitpoint here. store it.
    hit->dist = thit;
    hit->trinum = tri->triNum;
    hit->u = beta;
    hit->v = gamma;
    return ;
}
```

3.6 Summary

This chapter discussed the various issues associated with converting a CAD model into a suitable triangle format that can be used for ray-tracing. Most CAD model packages do not provided facet data in a useful format for EM interaction and so a method of converting the triangle soup into a triangle *mesh* was provided together with a technique that allows the surface of components withing the CAD model to have a roughening filter applied thereby approximating real-world scattering effects.

It was highlighted that an expensive component of ray-tracing is the calculation for ray-triangle interaction. An algorithm was presented that significantly reduces the computational cost of this calculation by first projecting the triangle into a two-dimensional plane and performing a coordinate conversion into barycentric space. This approach when coupled with a range of computing optimisations was able to reduce the computational complexity from approximately 267 compute cycles to approximately 20.

An efficient algorithm to determine whether a ray intersects with a triangle is only part of the story of ray-tracing. Another optimisation reduces the need to perform the ray-triangle intersection test on each triangle within the triangle mesh. This is the subject of teh next chapter.

Chapter 4

Accelerated Data Structures for Efficient Ray Tracing

Having discussed ray-triangle intersection in Chapter 3, the next most significant impediment to efficient ray-tracing is the identification of the primitives to test (Figure 4.1). If a ray entering a volume has to test against every primitive within that volume then the ray tracing routines can quickly become inefficient performing many unnecessary ray-primitive tests.

A simple way to reduce the primitive test is to use a parametric approach. This simple approach works well for ray tracers that support complex primitive types (such as parametric surfaces) and many costly ray-primitive objects can be avoided by tightly enclosing these primitives with *bounding volumes* [113, 114]. A bounding volume is a simple geometric primitive (usually a box or a sphere) that can be intersected very quickly. If a ray misses the bounding volume (which can be checked quite cheaply), it does not have to be intersected with the complex primitive at all. Only rays hitting the bounding volume have to be checked also against the original primitive. Bounding volumes are a standard-technique in ray tracing, but unfortunately they pay off only for complex primitive types. Even though the work in this thesis is focused on triangle primitives, it uses bounding volumes for initial testing. Note that there is a difference between Bounding Volumes and Bounding Volume Hierarchies for hierarchical scene subdivision (see below). As the calculation for intersection with a bounding volume is such an intrinsic part

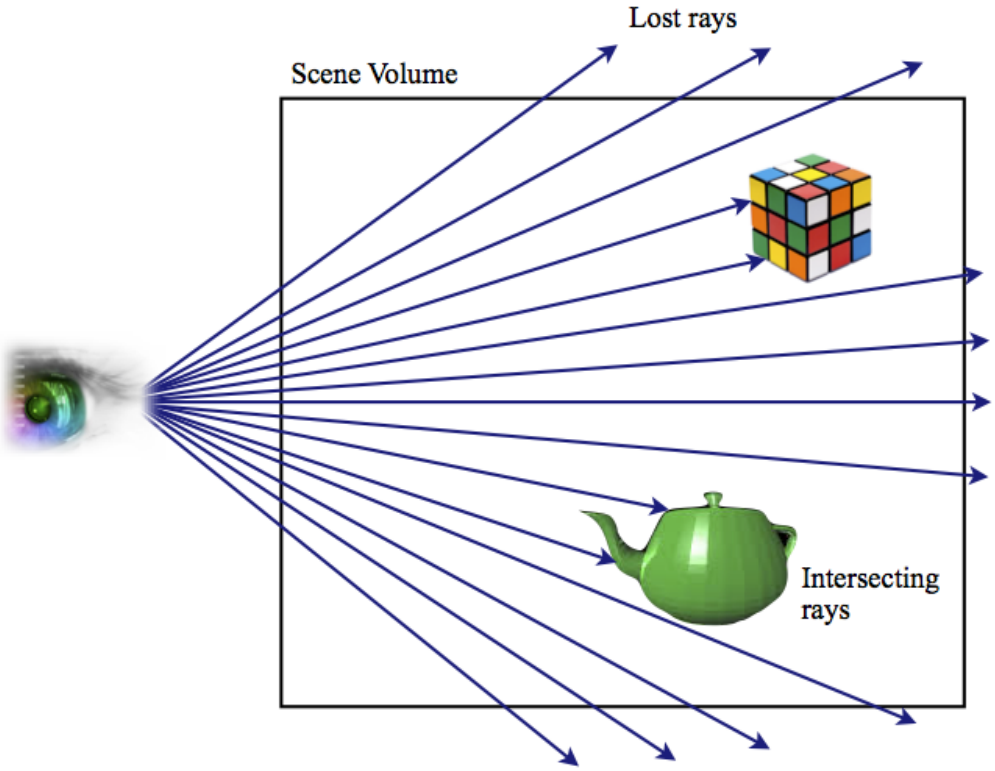


Figure 4.1: Checking each ray for intersection with each primitive within a scene volume is inefficient due to the number of rays that are lost and primitives that are no where near the ray's trajectory

of any ray-tracer, its efficiency in operation is of paramount importance. For this reason a technique was adapted for this work that will now be described in detail.

4.1 Checking for Bounding Volume Intersection

A key component of accelerated ray tracing is the implementation of a simple routine to check whether a ray has intersected a volume of space. In order to reduce complexity and, more importantly to remove trigonometric calculations, a common practice is to use axis aligned bounding box (AABB) (or sometimes ABs). These boxes bound primitives within the scene in a volume whose principal axes are aligned to the Cartesian grid. The technique implemented here is an extension of the original bounding box clipping algorithm [115]. This technique was developed by Danny Cohen and Ivan Sutherland in 1967 to efficiently determine how to clip lines to a view-port for a flight simulator project [116]. It gains its efficiency through the use of lookup codes (or OutCodes) to determine whether a point is within or

Type	Name	Value	bit sequence
INT	INSIDEAABB	0	000000
INT	LEFTAABB	1	000001
INT	RIGHTAABB	2	000010
INT	NEARAABB	4	000100
INT	FARAABB	8	001000
INT	TOPAABB	16	010000
INT	BOTTOMAABB	32	100000

Table 4.1: Modified *OutCodes* for a 3D Cohen-Sutherland Clipping Algorithm

outside a bounding box. The codes (here extended from the original work to the three dimensional case) are provided by Table 4.1:

These codes are assigned to the start and end points of the ray when compared to a bounding volume which can then be included or excluded based upon whether:

- Both endpoints are in the bounding volume (bitwise OR of endpoints equals 0) : trivial accept;

- Both endpoints are outside the bounding volume, but in the same region of space. (bitwise AND of endpoints does not equal 0): trivial reject

- Both endpoints are in different regions: In case of this nontrivial situation the algorithm finds one of the two points that is outside the volume (there will be at least one point outside). The intersection of the outside point and extended volume border is then calculated (i.e. with the parametric equation for the line) and this new point replaces the outside point.

- The algorithm repeats until a trivial accept or reject occurs.

The algorithm for calculating the OutCodes can be seen in Algorithm 9 with the associated Cohen-Sutherland clipping routine provided in Algorithm 10. This algorithm is very efficient as the simplest cases, where the ray is either completely within or outside the volume are quickly accounted for through bitwise operations.

Algorithm 9 Compute the bit code (OutCode) for a point $p(x,y,z)$ using the clip volume bounded diagonally by the 3D coordinates $min(x,y,z)$, and $max(x,y,z)$

```
1:  procedure ComputeOutCode( p, min, max)
2:      OutCode code
3:      code ← INSIDEAABB
4:      if p.x < min.x then
5:          code ← (code or LEFTAABB)
6:      else if p.x > max.x then
7:          code ← (code or RIGHTAABB)
8:      if p.y < min.y then
9:          code ← (code or NEARAABB)
10:     else if p.y > max.y then
11:         code ← (code or FARAABB)
12:     if p.z < min.z then
13:         code ← (code or BOTTOMAABB)
14:     else if p.z > max.z then
15:         code ← (code or TOPAABB)
16:     return code
```

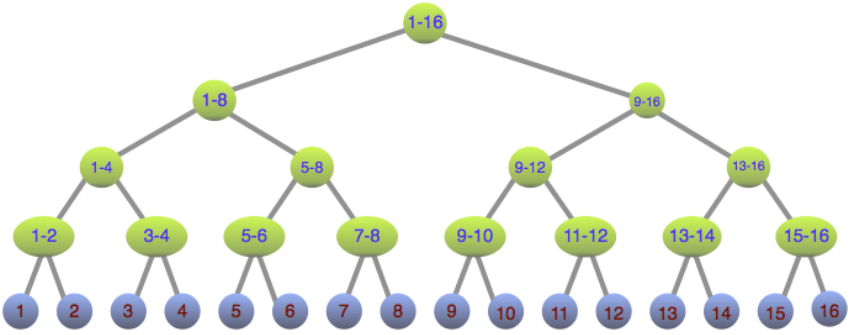
Algorithm 10 Clip a line between two points $p0(x,y,z)$ and $p1(x,y,z)$ against a volume $AABB(min,max)$

```
1:  procedure ClipToBox( pointer *p0, pointer *p1, min, max)
2:      outcode0 ← ComputeOutCode(p0,min,max)
3:      outcode1 ← ComputeOutCode(p1,min,max)
4:      if not (outcode0 or outcode1) then     ▷ line inside AABB. Trivial accept
5:          return true
6:      else if (outcode0 and outcode1) then   ▷ line outside AABB. Trivial reject
7:          return false
8:      else
9:          ▷ Line ends are outside AABB and line passes through AABB.
10:         ▷ Clip ends of line to volume AABB
11:         if outcode0 then
12:             p0 ← clip(p0, AABB(min,max))
13:         if outcode1 then
14:             p1 ← clip(p1, AABB(min,max))
15:     return true
```

4.2 Acceleration Structures for Scene Sub-Division

A successful way to accelerate ray tracing is to reduce the number of ray-primitive intersection operations. This is usually achieved by building an index data structure that allows for those primitives that are *close* to a given ray to be quickly found, and to skip primitives that are far away. During traversal of this data structure, only those potential candidates have to be tested for intersection, and the total number of intersection tests can be significantly reduced. Figure 4.2 describes why hierarchical data structures provide a performance increase (and one that can be measured against the theoretical limits of performance). Over the last 30 years, many different kinds of acceleration structures have been developed, e.g. uniform, non-uniform, recursive and hierarchical grids [117–121], Octrees [122–125], Bounding Volume Hierarchies, [113, 114, 126, 127], BSP trees (or k dimensional tree (kd-tree)s) [128–130], and even higher-dimensional, directional techniques such as ray classification [131, 132].

A Simple Sequence of Numbers. To determine whether a number (say 12) exists in this sequence requires it to be compared sequentially with the first 11 numbers before it can be confirmed. Some numbers will be earlier in the sequence whilst others are later. On average there will be $N/2$ comparisons. The *big-oh* $O()$ notation used in computer science for algorithmic complexity always assumes the worst case (where the number is at the end of the list in this example) and so a comparison with all other numbers in the list is required. It therefore takes, $O(N)$ operations for the worst case test.



A Binary Tree. To test whether a number is in this tree (say 12), only 4 comparisons are required, or more precisely, for a balanced tree, $O(log_2 N)$ comparisons. To insert N values into this tree therefore requires $O(Nlog_2 N)$ operations

Figure 4.2: Explanation of the computational improvement gained when hierarchical data structures (such as a binary tree) are used

As already noted by Kay [114] in 1986, the main difference between each of these techniques is whether they hierarchically organise the scene primitives (as done by Bounding Volume Hierarchies), or whether they subdivide object space (or ray space) into a set of unique voxels (as done by BSPs, kd-trees, or Grids).

4.2.1 Bounding Volume Hierarchies (BVH)

In the first class (Bounding Volume Hierarchies), each primitive is stored only once in the hierarchy. This usually leads to a predictable memory consumption, and guarantees that each primitive is intersected exactly once during traversal of the hierarchy (which is usually not the case for spatial subdivision techniques, see below). On the other hand, different parts of the hierarchy may overlap the same regions in space. This often leads to inefficient traversal such as intersecting the primitives in the wrong order or traversing the same space multiple times.

4.2.2 Spatial Subdivision (Octrees, Binary Space Partitions, etc.)

In contrast to BVHs, spatial subdivision techniques subdivide three dimensional space into a finite, non-overlapping set of volume elements (voxels), in which each voxel keeps a list of references to all the primitives overlapping that respective voxel (Figure 4.3).

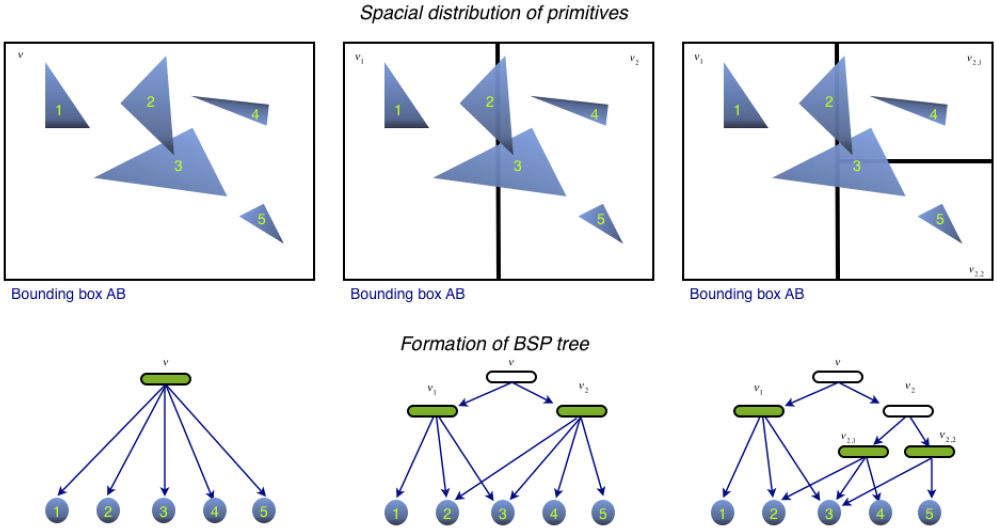


Figure 4.3: Spatial Subdivision; The scene volume is spatially divided into regions. Each region then forms part of a hierarchical tree. Only those nodes on the tree close to the ray trajectory are considered for further ray-primitive interaction.

Traversing a ray through the spatial data structure sequentially iterates through all the voxels encountered by a ray, thereby intersecting only those primitives referenced by an individial voxel (Algorthm 11 which is adapted from [133]). If this traversal is performed in front-to-back order, spatial subdivision techniques allow for early ray termination: As soon as a valid intersection is found at a certain distance along the ray, all the voxels behind this distance can be immediately skipped without any further computations. On the other hand, spatial subdivision has the draw-back that the number of encountered primitives is most reduced if the voxels get very small, which usually requires lots of memory. In that case, however, primitives will usually overlap many different voxels. This not only further increases the memory consumption of these techniques, but often leads to the same primitive being encountered several times during traversal of a ray. This is of particular importance in modern-day ray-tracing as many techniques now use the massively parallel architecture available through GPU processing. GPU's are excellent at providing fine-grain parallelisation (i.e. where the algorithm is small and essentially the same for all data[1]) but are severely limited by the amount of memory available.

Among all the published ray tracing optimisation techniques, hierarchical subdivision methods are the most important, as they allow the computational complexity to be reduced (from $O(N)$ to $O(Nlog_2N)$ [128] and Figure 4.2), whereas all other methods, roughly speaking, only improve *the constants* of the ray tracing algorithm. As such, any non-trivial ray tracing system today uses one or another form of the previously discussed scene subdivision and traversal techniques.

While hierarchical subdivision methods are commonly agreed to have the same computational complexity of $O(Nlog_2N)$, their respective performance in practice varies from case to case, depending on scene, implementation, hardware platform, and application (i.e. ray distribution). As such, it is not possible to name one single method that is always best, though practice has shown that K-dimensional trees (kd-trees) usually perform at least comparable to any other technique [128, 134]. Though much of the original ray tracing literature covers Octrees, in practice today the most

[1]Called Single Instruction Multiple Data or SIMD

Algorithm 11 Ray-tracing through a binary space partition data structure

```
1:  ▷ input: R = (O, D). Ray with origin O and direction D
2:  ▷ input: N ← root node of tree
3:  ▷ input: λ_entry ← Entry distance through tree for ray R
4:  ▷ input: λ_exit ← Exit distance through tree for ray R
5:  procedure TRAVERSE(N, R)
6:      while λ_entry < λ_exit do
7:          P_entry ← O + λ_entry D
8:          while N is empty do
9:              N ← { N_left−child   , if P_entry left of split
                    { N_right−child  , otherwise
10:         ▷ If here then found a node with something in it (a leaf node)
11:         ▷ Check for intersection againts all triangles in Node
12:         for all triangles T in N in front-to-back order do
13:             I ← INTERSECT(R, T)
14:             if I not NULL then
15:                 return (I, T)
16:         ▷ No intersection. Follow link to adjacent node and continue
17:         λ_entry ← EXITDISTANCE(R, N)
18:         P_exit ← O + λ_entry D
19:         N ← node adjacent to N at point P_exit
20:         if N = NULL then return (no intersection)
21:     return (no intersection)
```

commonly used techniques are either uniform *grids* or kd-trees. Grids (especially uniform ones) are advantageous because of their simplicity. Because of this, they are also well suited for implementation on hardware architectures with a restricted programming model (such as e.g. GPUs)). Kd-Trees on the other hand usually adapt better to varying scene complexity, and often achieve superior performance if used correctly. For this reason kd-trees (or more correctly axis-aligned binary space partitioning trees[2]) are used here.

[2]an important note taken directly from [110] clarifies the often confusing terminology associated with kd-trees:

> Quite commonly, axis-aligned BSP trees are also called *kd-trees*, or simply *BSPs* [129]. Unfortunately both terms are not ideally suited: *BSP* refers to the more general binary space partitioning tree with arbitrarily oriented splitting planes; and *kd-tree* also refers to a special form of higher dimensional binary trees for storing points [135,136]. Such kd-trees are also used e.g. for photon mapping [137], but are fundamentally different from axis-aligned BSP trees as used in ray tracing. As such, the term "axis-aligned BSP tree" is most exact, and thus preferable. Unfortunately, it is also quite awkward. In the remainder of this thesis, we will equally use all three terms -BSP, axis-aligned BSP, and kd-tree - but will always refer to the concept of axis-aligned BSP trees.

Except for faster intersection tests and improvements in hierarchical subdivision, other techniques (such as reducing the number of rays in one or another form) are limited to special cases and to a restricted set of applications. As such they are of limited use for a radar ray tracing system and will thus not receive any further attention in the remainder of this thesis.

4.3 Building Efficient Kd-Trees and Building Them Efficiently

There are two key components to building an efficient accelerated data structure for ray tracing. The first is the build time. Although not as critical as the time required to ray trace through the tree (as rebuilding the scene is not likely to be required as often as ray-tracing), it is important that the algorithm be streamlined as complex scenes with many millions of primitives can quickly become unwieldy. The second and perhaps more important component is that traversal over the nodes of the structure must be very fast so that primitives for ray-intersection testing are quickly identified.

Fortunately there are some excellent works in this area that are readily adapted for the purposes of this thesis. The principal work here is [135]. It draws heavily on many other works e.g. [112, 128, 136, 137] to provide a comprehensive summary of the current state-of-the-art in accelerated data structures (in this case kd-trees).

The intention here is to be able to build a kd-Tree in $O(Nlog_2N)$ operations but before a discussion on construction optimisation can start it is worth detailing the unique properties of a kd-tree.

4.3.1 Overview of a Kd-Tree

A Binary Space Partitioning (BSP) tree is a spatial subdivision that can be used to solve a variety of geometrical problems. It was initially developed as a means of solving the hidden surface problem in computer graphics [138]. It is a higher dimensional analogy to the binary search tree. The BSP tree has two major variants in computer graphics, which are call *axis-aligned* and *polygon-aligned*.

The polygon-aligned form [139, 140] chooses a plane underlying the polygon as the splitting entity that subdivides the spatial region into two parts. The scene is typically required to contain only polygons, which may be too restrictive for scenes with curved surfaces. For optimisation reasons, this work does not deal with the polygon-aligned form of BSP tree. For a survey and application of the techniques based on the polygon-aligned form of the BSP tree see for example [141, 142].

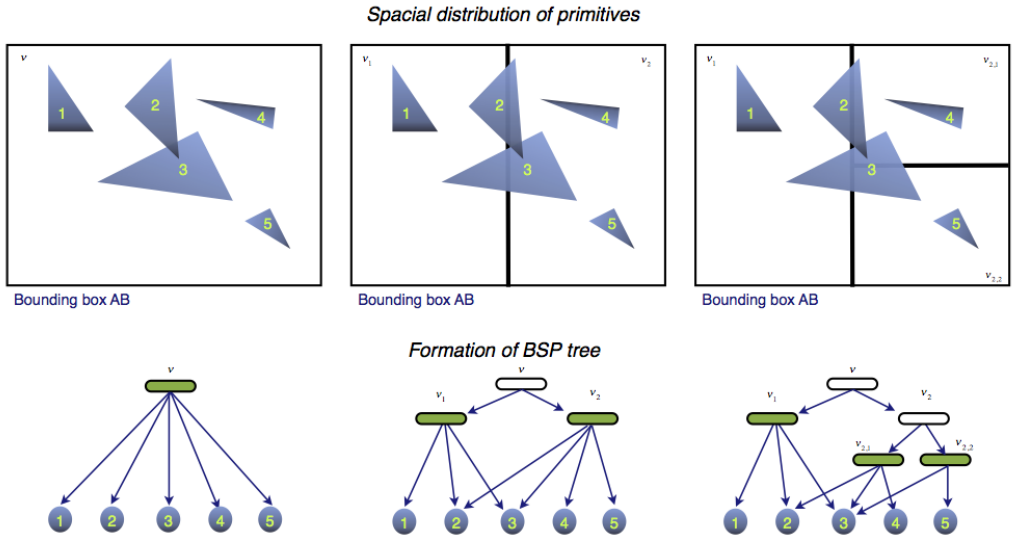


Figure 4.4: Building a Kd-tree. The bounding volume is partitioned spatially into nodes with leafs containing references to individual primitives that overlap with that node

The kd-tree is an axis-aligned form of the BSP tree in which the splitting entity is the plane that is always perpendicular to one of coordinate axes. The concept was first used for a Ray Shooting Algorithm (RSA) by Kaplan [143]. Since the splitting planes are perpendicular to the coordinate axes, the spatial subdivision is also called a rectilinear BSP tree or an orthogonal BSP tree.

A kd-tree tree for a set of objects S is defined as in Figure 4.4 and as follows: Each node v in the tree is associated with its axis-aligned bounding box $AB(v)$. The node associated with the root of the kd-tree is the axis-aligned bounding box of all objects from S. Each interior node v of the tree is assigned a splitting plane H_v that subdivides $AB(v)$ into two cells. Let H_v^+ be the positive half-space and H_v^- be the negative half-space bounded by $AB(v)$. The cells associated with the left (or negative) side of the node v are given by $AB(v) \cap H_v^-$ and the cells associated

with the right (or positive) side of node v are given by $AB(v) \cap H_v^+$. Each node v within the kd-tree is in itself a kd-tree. The left subtree contains the set of objects $S_v^- = \{s \cap H_v^- \neq 0 \mid s \in S_v\}$ and similarly the right subtree is defined by the set $S_v^+ = \{s \cap H_v^+ \neq 0 \mid s \in S_v\}$. Each interior node of the tree is called a *branch*, *node* or *sub-tree* according to the paper-du-jour. An exterior node, i.e. a node without any subsequent sub-trees is called a *leaf* and is often defined as v^E. Each leaf node contains a pointer to a list of objects S_{v^E} that intersect with the axis-aligned bounding box $AB(v^E)$ associated with the node v^E. A leaf that contains at least one primitive is called a *full leaf* and a leaf with no primitives is called an *empty leaf*. Therefore the axis-aligned bounding box associated with node v^E is called a leaf-cell defined by $AB(v^E)$. Papers often denote the left or negative sub-tree of node v as $lchild$ and the right or positive sub-tree is called $rchild$.

The kd-tree has a very useful attribute - its adaptability to the scene geometry caused by the ability to select the location of the split-plane for each level of the tree. This is most commonly selected as the mid-point of each axis [143] as it provides computational consistency and is easy to calculate. More recently however, performance increases have been demonstrated using a more complex form of split-plane selection based upon the surface area of each node and its children. This will be discussed in the next section.

4.3.2 Basic Kd-Tree Construction

As was seen in Figure 4.2, the most efficient way that a binary tree can be built is using $O(Nlog_2N)$ operations. In its simplest form, this requires no more than a recursive function call to construct each sub-node of the tree until a defined termination criteria is met. This can be seen in Algorithm 12.

There are several key components of Algorithm 12 that require careful consideration in order to approach maximal efficiency. First the TERMINATE function (line 5) has to usefully determine when to stop construction of the tree as calculation of splitting planes and sorting triangles into child nodes may be more expensive than searching the current node for ray-primitive intersections.

Secondly the requirement to find the splitting plane for each node using the

Algorithm 12 Pseudo-code to build a kd-tree using recursion

```
1:  ▷ input: Scene bounding volume AB. Triangle array T
2:  procedure BUILDKDTREE(T, AB)
3:      RETURN BUILDNODE(T, AB)
4:  function BUILDNODE(T, AB)
5:      if TERMINATE(T, AB) then
6:          RETURN LEAF
7:      SPLITPLANE p ← FINDSPLITPLANE(T, AB)
8:      AB⁻, AB⁺ ← SPLITNODE(AB, p)
9:      T⁻ ← TRIANGLESINVOLUME(T, AB⁻)
10:     T⁺ ← TRIANGLESINVOLUME(T, AB⁺)
11:     node_l ← BUILDNODE(T⁻, AB⁻)
12:     node_r ← BUILDNODE(T⁺, AB⁺)
13:     RETURN
```

FINDSPLITPLANE function on line 7 can be quite computationally expensive. As previously discussed, a simple solution to this is to mid-point split or median-split each node and proceed through the algorithm until each child node contains fewer than a predefined number of primitives. Whilst being efficient for kd-tree construction, it produces a data structure whose leaf nodes may be very large and not tightly bound to the facets within them. This means that upon ray traversal during the ray-tracing step of the simulator, many rays will be considered to intercept the leaf node but the probability of a ray intersecting a facet within that node are slight.

The solution to this is to better determine the location of the split planes such that they tightly bound the scene primitives and increase the probability of a ray intersecting a triangle once a leaf node is found. Fortunately a metric exists that helps to determine the optimal location of the split planes whilst assisting in the optimisation of the termination criteria. This is called the *surface area heuristic*.

4.3.3 The Surface Area Heuristic

The surface area heuristic (SAH) [136] can be used to split a voxel AB with splitting plane p into its sub-volumes AB^- and AB^+. Each volume, either side of the splitting plane has associated with it a number of primitives (triangles in this thesis) N_L and N_R. It then estimates the expected cost of traversing the newly split voxel in order to determine whether it should proceed any further. From [136], the SAH algorithm

makes the following assumptions:

- Rays are uniformly distributed, infinite lines, i.e.. They neither start nor terminate nor get blocked inside a voxel.
- The cost for both a traversal step of the kd-tree and the cost for a triangle intersection are known and are κ_T and κ_I respectively.
- The cost of intersecting N triangles is approximately $N\kappa_t$ i.e. Directly linear to the number of triangles.

From geometric probability theory [144] the conditional probability P of hitting a sub-voxel $AB_{sub} \subset AB$, assuming uniformly distributed rays, is:

$$\mathrm{P}_{[AB_{sub}|AB]} = \frac{SA\left(AB_{sub}\right)}{SA\left(AB\right)} \tag{4.1}$$

Where $SA(AB)$ is the surface area of the axis-aligned bounding box AB. The expected cost $Cv(p)$ for a given plane p can now be calculated as the sum of the additional traversal step plus the expected cost of intersecting each of the new children:

$$C_v\left(p\right) = \kappa_T + \mathrm{P}_{[AB_L|AB]}C\left(AB_L\right) + \mathrm{P}_{[AB_R|AB]}C\left(AB_R\right) \tag{4.2}$$

Expanding this provides the solution to determining the optimal kd-tree:

$$C\left(T\right) = \sum_{n \in nodes} \frac{SA\left(AB_n\right)}{SA\left(AB_S\right)}\kappa_T + \sum_{l \in leaves} \frac{SA\left(AB_l\right)}{SA\left(AB_S\right)}\kappa_I \tag{4.3}$$

Where AB_S is the scene bounding box AB_n is the bounding box for each node in the tree and AB_l is the bounding box for a leaf (a node with no children) within the tree.

In other words, when equation 4.3 is minimised then the kd-tree is the least costly and therefore traversal across it will be the fastest that could be achieved. Determining values for this equation would therefore provide the solution to determining the optimal kd-tree for traversal. Unfortunately the number of possible trees (nodes and leaves) for a given scene rapidly becomes unwieldy for all but the simplest of scenes and so this globally optimal tree is considered infeasible to

implement [135] for any realistic scene size. As a result an approximation can be made that assumes that each node is *locally greedy*. This means that the cost of subdividing a volume AB with plane p is calculated assuming that each of the child nodes are made into leaves. In this case each child node would contain triangles and so the cost can be estimated using:

$$C_v(p) \approx \kappa_T + \mathrm{P}_{[AB_L|AB]} |T_L| \kappa_I + \mathrm{P}_{[AB_R|AB]} |T_R| \kappa_I \tag{4.4}$$

$$= \kappa_T + \kappa_I \left(\frac{SA(AB_L)}{SA(AB)} |T_L| + \frac{SA(AB_R)}{SA(AB)} |T_R| \right) \tag{4.5}$$

where T_L is the number of triangles to the left of the split plane, T_R, the number of triangles to the right of the split plane and AB_L and AB_R are the bounding volumes for the left and right child nodes respectively.

This approach overestimates the cost of a given subdivision because many nodes can in actuality be subdivided further therefore reducing the total cost of attempting to intersect each triangle within a node. In practice however, it has been shown [135] that although many theoretically better approaches have been suggested no consistently better approximations have been found.

4.3.4 Optimal Split Plane Selection

Having developed a way to estimate the computational cost of two sub nodes separated by a split-plane p (given AB^-, AB^+, N_L and N_R), the next task is to search across a volume to find the split plane that would yield the lowest traversal and intersection cost. As there are an infinite number of plane positions p that could be selected, a more practical approach is required.

Consider Figure 4.5. In this figure triangles exist only in the x-y plane and only planes aligned to the y-axis are being considered. As the split plane p_y is swept along the x-axis, the cost function is calculated for both the left and right sides of the split plane and shown as the top line on the lower graph. As equation 4.5 suggests, this cost is a function of both the number of triangles on each side of the plane (lines with triangles in bottom graph) and the *surface area* of the volumes on each side of the split plane (in this 2D case this is just the area and show as the two solid

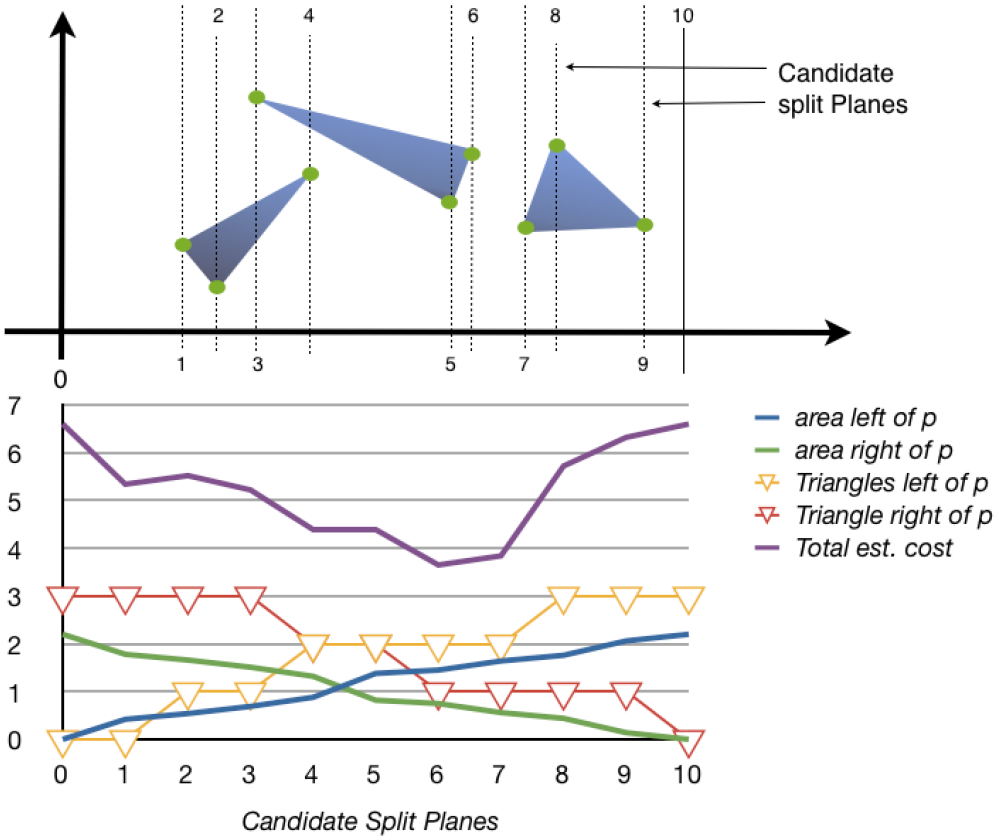


Figure 4.5: Representation of SAH estimated cost for split candidates across a two dimension volume. It can be seen that the cost function (top line) only changes at points corresponding to triangle vertices. Rather than search across infinite numbers of split planes, only the planes passing through vertices (called candidate planes) need to be considered.

lines). When viewed like this it becomes clear that the cost function only transitions at positions corresponding to triangle vertices. As a result only those planes passing through triangle vertices need to be considered. In the example above the minimum cost function occurs at split plane candidate index 6 which would then be selected as the plane to divide the bounding volume into the two sub volumes.

4.3.5 Automatic Termination Criteria

Another useful attribute to the SAH algorithm is that it naturally provides a metric by which further subdivision of a branch of the tree can be terminated. If the cost of the cheapest subdivision is higher than the cost of intersecting a ray with the

triangles within the current, undivided volume, then subdivision should stop:

$$\textsc{Terminate}(T,AB) = \begin{cases} true & \textsc{min}\left(C_v\left(p\right)\right) > \kappa_I\left|T\right| \\ false & \text{otherwise} \end{cases} \tag{4.6}$$

As described above this approach uses a *locally greedy* approximation which means that the cost of subdivision can be estimated as being higher than it actually would be. This most commonly occurs when a split candidate would cut off empty space (e.g. split plane 9 or split plane 1 in Figure 4.5) as one of the sub-volumes associated with these planes would have zero triangles and therefore sub-trees originating from one of these volumes would have zero cost. A commonly used modification to the SAH cost function [128, 138] is to bias the cost function by multiplying by a constant if either side of the split plane contains no triangles:

$$\lambda\left(p\right) = \begin{cases} 80\% & \left|T_L\right| = 0 \vee \left|T_R\right| = 0 \\ 1 & \text{otherwise} \end{cases} \tag{4.7}$$

4.3.6 Triangle Classification

On face value calculating the number of triangles on each side of a split plane seems simple. It needs to be handled with care though as some triangles, especially in architectural scenes (i.e. with lots of right-angled sides) have axis aligned triangles within them. As a result some split planes will be aligned with a triangle and may contain the triangle within it. This *event* needs to be differentiated from the more common situation where a triangle vertex is just touching a split plane. In this case the split plane will never contain an intersection with the triangle. There are therefore three exact solutions for sorting triangles around a split plane:

$$\begin{aligned} T_L &= \left\{t \in T \middle| \textsc{area}\left(t \cap \left(AB_L \backslash p\right)\right) > 0\right\} \\ T_R &= \left\{t \in T \middle| \textsc{area}\left(t \cap \left(AB_R \backslash p\right)\right) > 0\right\} \\ T_P &= \left\{t \in T \middle| \textsc{area}\left(t \cap p\right) > 0\right\} \end{aligned} \tag{4.8}$$

With T_p as the number of triangles that are within the plane and where $AB_L \backslash p$ is the part of the volume AB_L that does not intersect the plane p. Having determined the contents of these sets, equation (4.5) can be used to determined the estimated cost for each split plane candidate. More correctly however the equation should be evaluated twice: firstly by combing $T_p$with T_L and secondly by combining T_p with T_R and then selecting the lowest cost of the two.

Algorithm 13 The Surface Area Heuristic cost function given a split plane position p along dimension d and a number of triangles to the left of, right of and on the split plane N_L, N_R, N_p respectively.

```
1:  procedure SAHCOST(AB_v, p, d, N_L, N_R, N_p)
2:      ▷ returns the cost of the split and the cheapest side
3:      V_l ← left part of AB_v split along dimension d at point p
4:      V_r ← right part of AB_v split along dimension d at point p
5:      empty ← not (N_p = 0 and (ISPLANAR(V_l, d) or ISPLANAR(V_r, d)))
6:      P_L ← SURFACEAREA(V_l)/SURFACEAREA(AB_v)
7:      P_R ← SURFACEAREA(V_r)/SURFACEAREA(AB_v)
8:      cost_L ← COST(P_L, P_R, N_L + N_p, N_R, empty)
9:      cost_R ← COST(P_R, P_R, N_L, N_p + N_R, empty)
10:     if cost_L < cost_R then
11:         cheapside ← LEFT, cost ← cost_L
12:     else
13:         cheapside ← RIGHT, cost ← cost_R
14:     else RETURN cost, cheapside

15: function COST(P_L, P_R, N_L, N_R, weightEmptySpace)
16:     if (N_L = 0 or N_R = 0) and weightEmptySpace then λ ← 0.8
17:     else λ ← 1
18:     RETURN (λ * (κ_T + κ_i (P_L N_L + P_R N_R)))
```

The surface area heuristic cost function can be calculated using Algorithm 13. In this algorithm the cost of a given split plane is returned to the calling function in addition to the cheapest side of the split plane. This is used later on to ensure that any triangles on the plane are combined with those in the lowest cost child.

A procedure to split a bounding volume based on the lowest cost split plane can now be considered and is presented in Algorithm 14.

This algorithm simply iterates first over all the triangles in the volume and calculates the split plane candidates for each. Then for each split plane it classifies

Algorithm 14 A simple way of splitting an axis-aligned bounding box using the surface area heuristic

```
 1: procedure SAHSIMPLE(TRIANGLES T, AABB AB)
 2:     for t ∈ T do
 3:         Cost ← ∞; P_cheap ← 0
 4:         P ← SPLITCANDIDATES(t, AB)        ▷ returns candidate split planes for t
 5:         for p ∈ P do
 6:             AB_L, AB_R ← SPLIT(AB, p)
 7:             T_l, T_p, T_r ← CLASSIFYTRIANGLES(T, AB, AB_L, AB_R, p)
 8:             cost, side ← SAHCOST(AB, p, p.dim, T_l, T_r, T_p)
 9:             if cost < Cost then Cost ← cost ; P_cheap ← p ; cheapside ← side
10:     AB_L, AB_R ← SPLIT(AB, P_cheap)
11:     T_l, T_p, T_r ← CLASSIFYTRIANGLES(T, AB, AB_L, AB_R, P_cheap)
12:     if cheapside = LEFT then RETURN (P_cheap , T_l ∪ T_p , T_r )
13:     if cheapside = RIGHT then RETURN (P_cheap , T_l , T_r ∪ T_p )

14: function CLASSIFYTRIANGLES(T, AB, AB_L, AB_R, p)
15:     T_l, T_p, T_r ← 0
16:     for t ∈ T do
17:         if t lies in plane p and (AREA(p ∩ AB) > 0) then
18:             T_p = T_p + 1
19:         else
20:             if AREA(t ∩ (AB_L \ p)) > 0 then T_l = T_l + 1
21:             if AREA(t ∩ (AB_R \ p)) > 0 then T_r = T_r + 1
22:     RETURN T_l, T_p, T_r
```

each of the triangles to be either to the left, right or within the split plane. Finally the cost for each split plane is calculated and the lowest cost returned. The algorithm also includes checks for zero volume bounding boxes (lines 17, 20 and 21) which are troublesome and can result in T_l, T_p and T_r having values of zero which must be accounted for in the calling function.

The cost of looping over each of the triangles is $O(N)$ and then for each split plane candidate each triangle has to be classified (another $O(N)$) which means that this simple algorithm is of the order of $O(N^2)$. As even a modest scene size may consist of a million triangles, this algorithm quickly becomes challenging.

Fortunately more efficient algorithms exist. Good examples are those provided by [145] and [146] which uses the same terminology as defined in [135]. They have been adopted in order to continue the story.

4.3.7 Kd-Tree Construction in *O(NlogN)*

From inspection of Algorithm 14, it is obvious that the main reason for the algorithm's inefficiency is that the triangles have to be repeatedly tested for their position relative to candidate split planes (themselves derived from the vertices of each triangle). As the vertices of the triangles do not change, a better approach is to take the triangle classification routine outside of the main loop. This is achieved by *sweeping* the splitting plane over all candidate positions (for each dimension k) and storing the results.

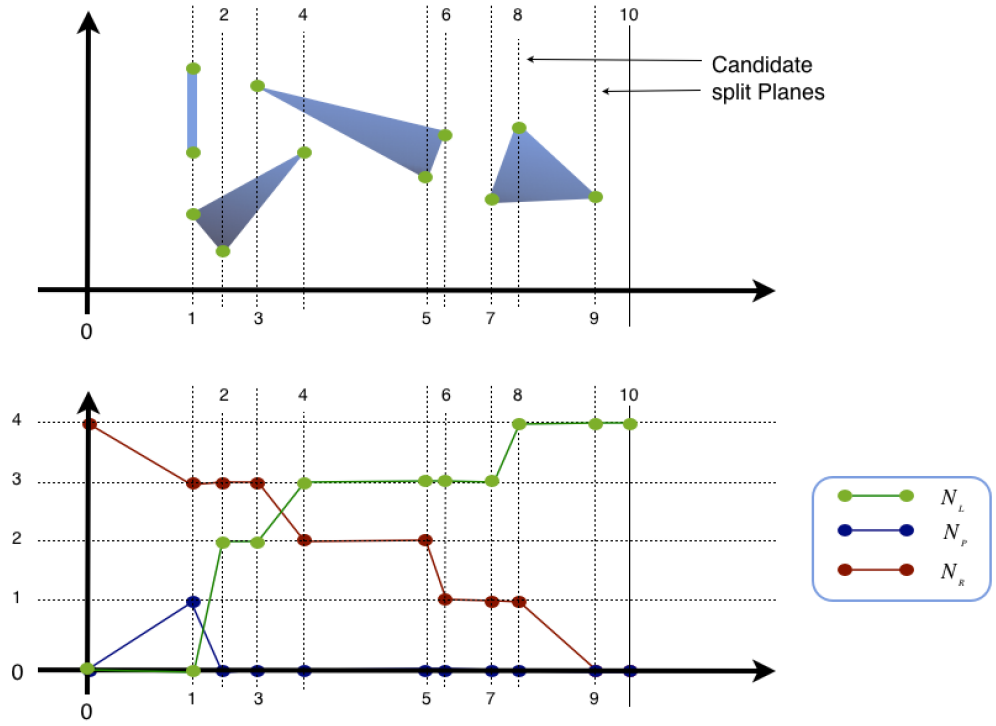


Figure 4.6: By sweeping the split plane incrementally across all split plane candidates, the number of triangles to the left of the split plane (green), on the split plane (blue) and right of the split plane (red) can be maintained without having to recompute.

Consider that for each split plane candidate p_k for a given dimension k there exists a certain number of triangles that are either starting, in the plane or ending. These are referred to as p^+, $p^|$ and p^- respectively. If only one dimension is considered at a time and the triangle vertices are stored in order within that dimension, then the number of triangles to the left of the plane, N_L, in the plane N_p and right of the plane N_R can be computed incrementally for each step of the plane sweep

(Figure 4.6). For example consider split plane 1 in Figure 4.6:

$$N_L^{(1)} = 0 \quad N_P^{(1)} = 1 \quad N_R^{(1)} = 3 \tag{4.9}$$

or, for N triangles:

$$N_L^{(1)} = 0 \quad N_P^{(1)} = p_1^{|} \quad N_R^{(1)} = N - p_0^{|} \tag{4.10}$$

It therefore follows that as the plane sweeps from p_{i-1} to p_i, the number of triangles, N_L, N_R and N_p will change as follows:

1. The new N_p will be $p_i^{|}$; these $p_i^{|}$ triangles will no longer be in AB_R. The triangles that were on plane $p_{i-1}^{|}$ will now be in AB_L
2. Those triangles that started at $p_{i-1}^{|}$ will now overlap AB_L
3. Those triangles ending at p_i will no longer overlap AB_R

This now develops three equations that define how the triangle count should be updated:

$$\begin{aligned} N_L^{(i)} &= N_L^{(i-1)} + p_{i-1}^{|} + p_{i-1}^{+} \\ N_R^{(i)} &= N_R^{(i-1)} - p_i^{|} - p_i^{-} \\ N_P^{(i)} &= p_i^{|} - 1 \end{aligned} \tag{4.11}$$

These can now be incorporated into the simple SAH algorithm by using them to first sweep across the volume AB in each dimension and calculating the properties at each split plane candidate. The plane candidates are called *events* and the algorithm proceeds as follows. First only one dimension k is considered. For this k all triangles within the volume t are considered. Triangles that are partly outside the volume are clipped to the volume using the modified Cohen-Sutherland algorithm described earlier (Algorithm 10. For each triangle the split plane candidates (or events) are identified as being either a starting event $\varepsilon_{type}(t,+)$, a planar event $\varepsilon_{type}(t,|)$ or an end event $\varepsilon_{type}(t,-)$. Each event, once calculated therefore consists of three parameters; the triangle that the event was triggered by, the position along the axis of the split plane and the type: $\varepsilon_k\left(\varepsilon_t, \varepsilon_p, \varepsilon_{type}\right)$.

When all events are calculated they are then sorted into an event list E by ascending position along the axis p. It frequently occurs that several events coincide in which case they are sorted such that end events precede planar events which precede start events. For algorithmic purposes this can be written for two events a and b as :

$$a < b = \begin{cases} true & \left(a_p < b_p\right) \vee \left(\left(a_p = b_p\right) \wedge \left(\tau(a) < \tau(b)\right)\right) \\ false & otherwise \end{cases} \tag{4.12}$$

Where $\tau\left(e_{type}\right)$ is 0 for end events, 1 for planar events and 2 for start events.

All that is now required to make the simple SAH algorithm more efficient is to use the *plane-sweep* technique to find the best split plane. This can be seen in Algorithm 15. There is a slight difference in this algorithm in the way that N_L, N_p and N_R are calculated. From Algorithm 13 the algorithm to calculate the SAH cost relies on the number of triangles left, right and in the split plane. As this function is called within a loop in Algorithm 15 then the code can be made slightly simpler by calculating the values of N_L after the first call to SAH.

This plane sweep technique effectively takes the requirement for sorting the event planes outside of the main loop. The next consideration however is that, once events have been sorted, they need to be maintained for each sub-division of the current AB. If they are not then, every time the current volume is divided then the event planes will have to be sorted again and so the computation will again be $O(N^2)$.

Fortunately there are a few observations that allow the event list to be maintained without sorting, even when a volume is subdivided:

- Triangles can be classified into T_L and T_R with only iterations over the event list and that the number of operations is $O(N)$ (see Algorithm 16)
- As the volume event list is already sorted then each sub-volume list will already be sorted.
- Triangles completely on one side of the split plane will not effect the ordering

Algorithm 15 Finding the best split plane by incrementally sweeping over all candidate split planes

```
1:  ▷ E is the set of all events in the volume AB sorted by dimension k
2:  procedure SAHPLANESWEEP(E, AB)
3:      N_L, N_R ← 0, N_R ← number of triangles in AB
4:      Cost ← ∞ ; P_cheap ← 0
5:      for p ∈ E do
6:          p^-, p^|, p^+ ← 0
7:          if ε_type = end then p^- = p^- + 1
8:          if ε_type = planar then p^| = p^| + 1
9:          if ε_type = start then p^+ = p^+ + 1
10:         N_p = p^|
11:         N_R = N_R − p^|
12:         N_R = N_R − p^-
13:         cost, side ← SAHCOST(AB, p, k_p, N_L, N_R, N_p)
14:         if cost < Cost then
15:             Cost ← cost; P_cheap ← p
16:             cheapside ← side; k_cheap ← k
17:         N_L = N_L + p^+
18:         N_L = N_L + p^|
19:     if Cost > κ_I N_triangles then
20:         set this node to be a leaf and RETURN
21:     else
22:         RETURN (P_cheap, k_cheap, cheapside)
```

of the event list after the split.

- Triangles straddling the split plane will require additional clipping which may generate more events. (see Figure 4.7)

- Splicing new events into an event list can be performed in $O(N)$ operations using a single MERGESORT iteration provided the event lists are ordered.

Having classified the triangles within the current volume as being either LEFT, RIGHT or BOTH, (i.e. left of the splitting plane, right of the splitting plane or on both sides of the splitting plane), the event array must now be split according to :

$$T_L = T_{Left} \wedge T_{Both} \quad T_R = T_{Right} \wedge T_{Both} \tag{4.13}$$

This is performed using Algorithm 17 which first separates the events into arrays

Algorithm 16 Splitting the triangles in volume AB in $O(N)$

```
1:  procedure SPLIT-TRIANGLES(E, k, p, cheapside)
2:      for e ∈ E do
3:          t ← e_triangle
4:          add t to T_both
5:      for e ∈ E do
6:          t ← e_triangle
7:          if e_type = END and e_k = k and e_p ≤ p then
8:              add t to T_left
9:          else if e_type = START and e_k = k and e_p ≥ p then
10:             add t to T_right
11:         else if e_type = PLANAR and e_k = k then
12:             if e_p < p or (p = e_p and cheapside = LEFT) then
13:                 add t to T_left
14:             if e_p > p or (p = e_p and cheapside = RIGHT) then
15:                 add t to T_right
16:     RETURN (T_left, T_right, T_both)
```

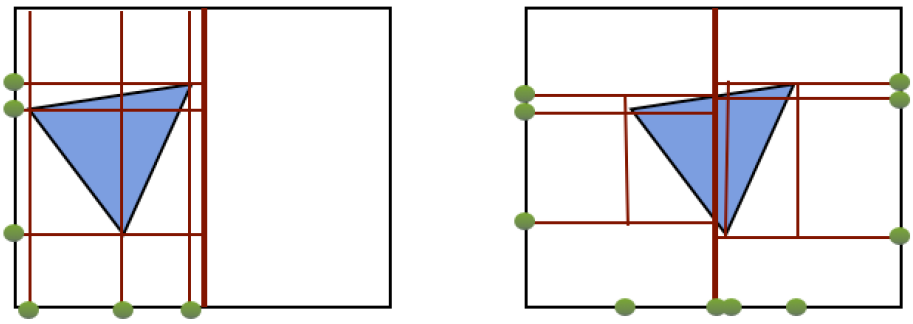


Figure 4.7: Triangles completely on one side of the split boundary maintain their events after subdivision (left). Triangles straddling the split plane (right) have to be clipped to the new sub-volume. This generates new events that have to be spliced into the ordered event list on both sides of the split plane.

E_{left} ,E_{right}. Those events that are associated with triangles straddling the split line have their triangles clipped to the volumes on either side of teh split and their events recalculated into two additional arrays $E_{bothleft}$ and $E_{bothright}$. The events in E_{left} can then be merge sorted with $E_{bothleft}$ in a single $O(N)$ pass (as their are each already sorted). The same applies to those events in E_{right} and $E_{bothright}$.

Algorithm 17 Splitting the triangles in volume AB in $O(N)$

```
1:  procedure SPLICEEVENTS(T_left, T_right, T_both)
2:      for e ∈ E do
3:          t ← e_triangle
4:          if e_type = LEFT then
5:              E_left ← e
6:          else if e_type = RIGHT then
7:              E_right ← e
8:          else
9:              V_l, V_r ← SPLIT(AB, p, k)
10:             E_bothleft ← CALCULATEEVENTS(V_l, t)
11:             E_bothright ← CALCULATEEVENTS(V_r, t)
12:     E_l ← MERGESORT(E_left, E_bothleft)
13:     E_r ← MERGESORT(E_right, E_bothright)
14:     RETURN E_l, E_r, (T_left ∪ T_both), (T_right ∪ T_both)
```

As the event sweep has now been extracted from the main SAH loop and maintained in $O(N)$ operations then the final algorithm can now be reconstructed to form a kd-tree in $O(Nlog_2N)$ operations. This can be seen in Algorithm 18. In this routine each node of the kd-tree is treated as a single object with the BUILDNODE routine being a recursive member function that accesses the parameters of the node. This routine calls the algorithms previously described in 15, 16 and 17 and uses the surface area heuristic to efficiently determine the lowest cost split plane for the sub-division. It also determines whether the tree should terminate as a leaf because the cost of further sub-division is greater than the predicted cost of ray intersecting the triangles within the current node.

4.3.8 GPU Optimisation

Building a kd-tree in $O(Nlog_2N)$ is theoretically the optimal efficiency. Speedups can still be achieved however through different implementation approaches and

Algorithm 18 Building a kd-tree in $O(Nlog_2N)$ operations

```
1:  ▷ input: Scene bounding volume AB. Triangle array T
2:  procedure BUILDKDTREE(T, AB)
3:      CALCULATEEVENTS(AB, T)
4:      RETURN BUILDNODE(T, E, AB)
5:  function BUILDNODE(T, E, AB)
6:      P_cheap, k_cheap, cheapside ← SAHPLANESWEEP(E,AB)   ▷ find best split plane
7:      if P_cheap denotes this node is a leaf then
8:          RETURN LEAF
9:      else
10:         T_left, T_right, T_both ← SPLIT-TRIANGLE(E, k, P_cheap, cheapside)
11:         E_l, E_r, T_l, T_r ← SPLICEEVENTS(T_left, T_right, T_both)
12:         AB_L, AB_R ← SPLIT(AB, k, P_cheap)
13:         BUILDNODE(T_l, E_l, AB_l)
14:         BUILDNODE(T_r, E_r, AB_r)
15:     RETURN
```

different processing architectures. One such architecture is the GPGPU which will now be discussed.

Contrary to common understanding, GPGPUs are not universally efficient. They are optimised for algorithms that have a large number of simple operations - often called *single instruction multiple data* (SIMD) problems. Algorithm 18 being a recursive algorithm is unsuitable for GPGPU programming[3] as such algorithms require large amounts of heap memory and the ability to fork a large number of processes.

The first publication to address real-time kd-tree construction on a GPU was [147] and their work has been used heavily here to evolve the kd-tree construction algorithm for SARCASTIC.

The first step in translating Algorithm 18 to the GPU is to replace the recursive element with an array of nodes that can reference each other. This is often called a *breadth-first* implementation whereas the recursive kd-tree algorithm is called a *depth-first* implementation. In the breadth-first implementation, each node at the same depth in the tree is considered in parallel. Child nodes are processed and generate a new array that is added to the previous one. By doing this, each call to the

[3]GPGPUs have only recently supported recursion on NVIDIA architectures with compute capabilities of 2.0 or greater. As of writing this OpenCL does not support recursion

GPU is more likely to apply the same instruction set to the different data structures of the tree and therefore maximise the GPU's utilisation. The basic breadth-first algorithm is presented in Algorithm 19.

Algorithm 19 Building a kd-tree in a breadth-first order

```
1:  ▷ input: Scene bounding volume AB. Triangle array T
2:  procedure BUILDKDTREEBREADTHFIRST(T, AB)
3:      initialise treelist
4:      thislevel ← new list with one node(T, AB)
5:      while thislevel not empty do
6:          nextlevel ← SPLIT(thislevel)        ▷ split all nodes at this level in tree
7:          treelist ← nextlevel
8:          thislevel = nextlevel
9:          CLEAR(nextlevel)
10:     RETURN treelist
```

Due to the sparse resources on a GPU, it is common to run the majority of Algorithm 19 on the CPU with the key components of the SPLIT routine (line 6) being farmed out to one or more GPUs.

The key innovation in [147] was to recognise that the cost of computing the surface area heuristic was quite expensive, especially higher up the tree where the nodes are larger and contain more triangles. The solution was to implement a two stage approach to building the tree with the larger nodes being split through a simple median splitting mechanism and only employing the surface area heuristic when required during the smaller nodes.

The correct implementation of this algorithm is challenging requiring the extensive use of GPU scan primitives [148]. Additionally, the authors did not use the standard 3D scenes provided by [149] and optimised their rendering times for 1kx1k images. The documented implementation of the algorithm is therefore challenging for performance comparison against other algorithms.

4.4 Performance Results of SARCASTIC's Kd-tree Builder

The requirement for SARCASTIC is to be able to rapidly rebuild the scene kd-tree so that the phase history of moving and vibrating targets can be simulated. Additionally, there are still many million more traversals through the kd-tree during the phase history generation and so it is critical to have a kd-tree that is precisely constructed and efficient to traverse. For this reason, the approach taken by [147] has been modified in SARCASTIC, the details of which will be covered here.

The main issue with [147] is that it does not recompute the correct number of triangles upon splitting the smaller nodes of the tree, choosing instead to accept that the SAH cost will be higher for each side of the split plane as the newly created nodes have not been correctly clipped to the triangles and no additional split candidate events generated. The reason provided for this is that it is more efficient on GPU memory as fewer smaller nodes are generated and that experiments have shown that "clipping rarely improves ray-tracing performance".

As SARCASTIC is highly dependent on efficient tree traversal, it was decided to implement the sort-free plane-sweep event recalculation described in Algorithm 4.6 at each level of the lower levels of the tree. The SARCASTIC implementation of the kd-tree construction is called FASTKDTREE and is provided in Algorithm 20.

Unfortunately the performance metrics in [147] were mainly provided in units of seconds and therefore are not that useful as CPU performance speeds constantly increase and GPU memory is ever expanding. The implementation of FASTKDTREE in SARCASTIC is therefore difficult to compare with [147] in any meaningful way. A more meaningful metric is provided in [135] that calculates the SAH traversal cost (and therefore the tree's efficiency) in terms of expected values for traversal, number of leaves visited and triangles intersected. The expected number of traversal steps can be computed from 4.3 :

$$E_T = \sum_{n \in nodes} \frac{SA\left(AB_n\right)}{SA\left(AB_S\right)} \tag{4.14}$$

Algorithm 20 Fast kd-tree construction engine in SARCASTIC

```
1:  procedure FASTKDTREE(T, AB, smallsize)
2:      thislevel ← new list with one node(T, AB)
3:      initialise treelist
4:      initialise smalllist
5:      while thislevel not empty do
6:          nextlevel, smalllist ← MEDSPLIT(thislevel, smallsize)      ▷ median split
7:          treelist ← nextlevel
8:          thislevel = nextlevel
9:          CLEAR(nextlevel)
10:     E ← CALCULATEEVENTS(smalllist)                                 ▷ build all plane events
11:     while smalllist not empty do
12:         nextlevel ← SPLITSMALL(smalllist, E)                       ▷ SAH split nodes
13:         E ← CLIPANDSPLICE(nextlevel, smalllist, E)                 ▷ Rebuild event list
14:         treelist ← nextlevel
15:         thislevel = nextlevel
16:         CLEAR(nextlevel)
17:     RETURN treelist
```

Similarly the expected number of leaves visited is provided by :

$$E_L = \sum_{l \in leaves} \frac{SA(AB_l)}{SA(AB_S)} \tag{4.15}$$

and finally the expected number of triangles intersected is :

$$E_I = \sum_{l \in leaves} N_n \frac{SA(AB_l)}{SA(AB_S)} \tag{4.16}$$

with N_n being the number of triangles in a given leaf node n. The total cost of a tree once built can be evaluated using Equation (4.3).

The algorithm provides a mechanism to account for memory limitations- the value of *smallsize* can be adjusted. This is the smallest number of triangles in a node before the algorithm transitions from median splitting to SAH calculations. When *smallsize* is very small then the algorithm performs very quickly as almost all the nodes are median-split. This has a disadvantage however in that the number of nodes significantly increases which increases both the memory requirements and therefore the number of expected traversals and hence the overall kd-tree cost. If *smallsize* is made larger then a greater number of the nodes towards the bottom

	Wald [135]	size of small node, *smallsize*							
Number of Triangles N_T	827k	2	4	8	16	32	64	128	256
Number of nodes N_n		315k	216k	170k	187k	145k	82k	44k	24k
Number of leaves N_L	1390k	157k	108k	85k	93k	73k	41k	22k	12k
Non empty leaves	627k	53k	75k	79k	93k	73k	41k	22k	12k
Ave tris/non-empty leaf	2.56	1	1.9	3.4	5.2	8.8	16.8	32.6	63.8
Expected Traversals E_T	76.5	37.5	37.5	38.2	40.4	41.1	39.5	37.5	35.4
Expected Leaves E_L	20.8	1.6	2.2	2.9	3.9	4.6	4.8	5	5.1
Expected Intersections E_I	8.4	0.7	3.3	9.3	19.2	38.3	76.7	153	303.9
Traversal Cost C_T	1316	577	629	758	991	1382	2127	3622	6608
build time (secs)	16	13.6	12.9	12.5	13.4	14.8	16.7	21.86	33.7

Table 4.2: Comparison of FASTKDTREE construction in SARCASTIC compared to [135]. The scene used is *The Stanford Dragon* taken from [149]

of the tree will be split using the SAH. This increases build time but reduces the number of nodes (and hence expected traversals). It also increases the number of intersection calculations required to find a ray-triangle intersection after a leaf node is found which can again reduce tree efficiency.

For comparison Table 4.2 highlights the trade-off and compares the performance of SARCASTIC with results published in [135]. Both papers used *The Stanford Dragon* [149] for performance evaluation (Figure 4.8) so that meaningful comparisons could be made.

It is interesting to note that the selection of *smallsize* is critical to both the build time and the final traversal cost of the tree. For this scene with *smallsize* set to 16 or higher the number of non-empty leaves remains the same as the number of nodes. This implies that the SAH algorithm is not effectively cutting away empty space. With a *smallsize* set to 8 or lower the number of empty leaves increases as the SAH algorithm starts to take over from the median splitting.

The results shown here suggest that the new algorithm is both faster and more efficient than previous kd-tree building algorithms. This is unlikely to be correct however as the algorithm is highly scene dependent. Through experimentation it has been found that certain scenes do not perform as well under the new approach compared to previous work. Upon closer inspection of these scenes, FASTKDTREE appears to perform well when the nature of the scene is distributed with large spaces and voids whereas it under-performs on such scenes where the facets are densely collected together such as *The Stanford Bunny* and *The Stanford Armadillo*

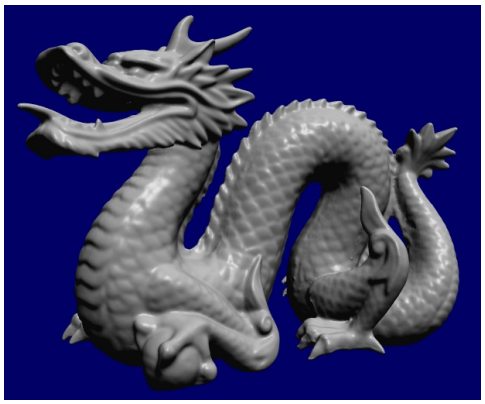


Figure 4.8: *The Stanford Dragon* taken from [149] and used for comparison of FASTKDTREE with [135]

(Figure 4.9). This is likely to be caused by a poor selection of *smallsize* and possible rounding errors caused by very small and densely packed facets which have not been correctly handled in the software. The results are encouraging though as the types of scenes likely to be ray-traced with SAR will have large open spaces and voids and have their facets distributed over a wide area.

Figure 4.9: (Left) *The Stanford Bunny* and (Right) *The Stanford Armadillo* are two freely available 3D triangle meshes [149] commonly used for comparison of kd-tree construction performance

For completeness, the FASTKDTREE approach was applied to a Main Battle Tank (Figure 4.10) and the intermediate axis-aligned bounding boxes saved so that the progress of the algorithm can be seen. When viewed as an animation it can be seen that the larger volumes are cut away quickly through the median-splitting routine and then the SAH algorithm starts to form bounding boxes tightly around the scenes

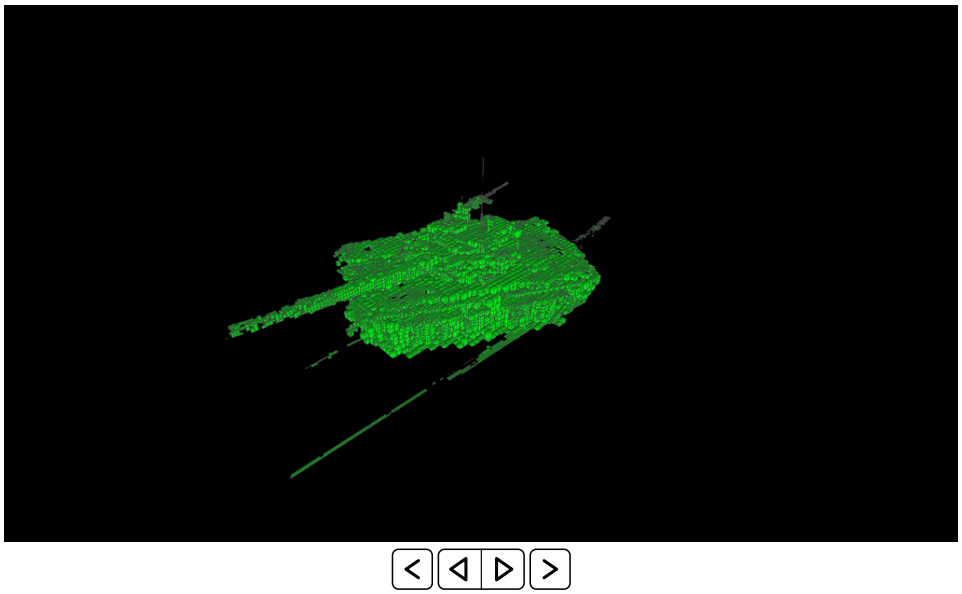


Figure 4.10: Animation showing the kd-tree construction of a Main Battle Tank scene by displaying axis-aligned bounding boxes for each level of the tree. The scene contains 3 million triangle facets[4]

facets.

The scene consists of approximately 3 million triangular facets and extends over a 10m x 10m area and took 64 seconds to construct the kd-tree. It is likely that this can be further improved both in terms of speed and kd-tree traversal cost. One area of improvement is related to the selection of *smallsize*. As previously discussed this is highly scene dependent although for a SAR simulation only small components of the scene will actually be moving. This means that a step prior to simulation may be performed to determine the optimal value for *smallsize*. Once identified, this can then be used throughout the rest of the simulation.

It now remains to integrate the FASTKDTREE algorithm into SARCASTIC and discuss final implementation optimisations. This is the subject of the next chapter.

4.5 Summary

This chapter completes a two-chapter treatise on ray-tracing which began in Chapter 3 with a discussion on triangles and the ray-triangle intersection calculation.

In this chapter, a further efficiency was described that reduced the need to

[4]This animation uses the LaTeXANIMATE plugin. If the animation is not playing then it is likely your PDF viewer does not support active content.

perform the ray-triangle intersection test on every triangle in the CAD model mesh. This efficiency is the bounding volume hierarchy.

A detailed discussion on bounding volume hierarchies was presented. There are two sides the construction of such a data structure: the efficient formation of the structure; and the structure itself being efficient for a ray to traverse in as few steps as possible. Both aspects were covered with algorithms for formation and ray traversal.

The chapter described previous work that identified the the surface area heuristic (SAH) is the most efficient measure to determine the maximum depth of the tree and an algorithm to generate such a tree in $O(Nlog_2N)$ operations was described. The algorithm attempts to tightly bound triangles and maximise empty space around them in order to reduce the number of traversal steps performed within the tree when ray tracing.

A new FASTKDTREE algorithm was then presented that improves the computational efficiency of generating such kd-trees by using the quicker *median-splitting* approach in the earlier stages of the tree and then transferring to the more burdensome but precise SAH algorithm to tightly bound the triangles. The algorithm was compared to published construction techniques in terms of tree-traversal efficiency and construction operations and shown to perform favorably.

At this stage in the thesis all the background material has been covered. This started with the calculation of scattered electromagnetic fields in Chapter 2. The thesis then proceeded to discuss the computer science aspects of CAD model mesh generation and ray-triangle intersection calculations in Chapter 3 before rounding off in this chapter with the efficient determination of which triangle within a simulated scene a radar ray interacts with.

The next chapter will bring this work together to build a SAR scene simulator called SARCASTIC that uses efficient radar ray-tracing techniques and scattered EM field calculations to simulate the expected EM signatures in such a way that a SAR image can be formed.

Chapter 5

Implementation of SARCASTIC

All the components for SAR simulation have now been discussed. This chapter will pull the related components together and describe the implementation of SARCASTIC. Verification and validation results will be provided.

5.1 Program Structure

SARCASTIC uses the UNIX philosophy of "*do one thing and do it well*" [150]. This means that the package is actually composed of many separate modules called one after the other, either from the command line or by calling a control script. Before delving into the workings of SARCASTIC itself it is worth first discussing the overall work flow. This can be seen in Figure 5.1.

There are two required inputs before SARCASTIC can be run. The first is a representation of the scene in terms of planar triangle facets. For ease of use and to avoid licensing issues, the preferred format is the Stanford .ply format [149]. CAD models are usually created using a user's preferred package (such as SketchUp® or Autodesk Maya®).

The second required input is a phase history dataset. The nature of phase history data was covered in Chapter 1 and is usually used as input into an image formation processor to convert the collected radar returned echoes into a SAR image. The preferred format for phase history data (PHD) is the *Common Phase History Data* [151] which provides a sensor agnostic way to represent collected SAR data without concern for the collection platform. The original concept for SARCASTIC

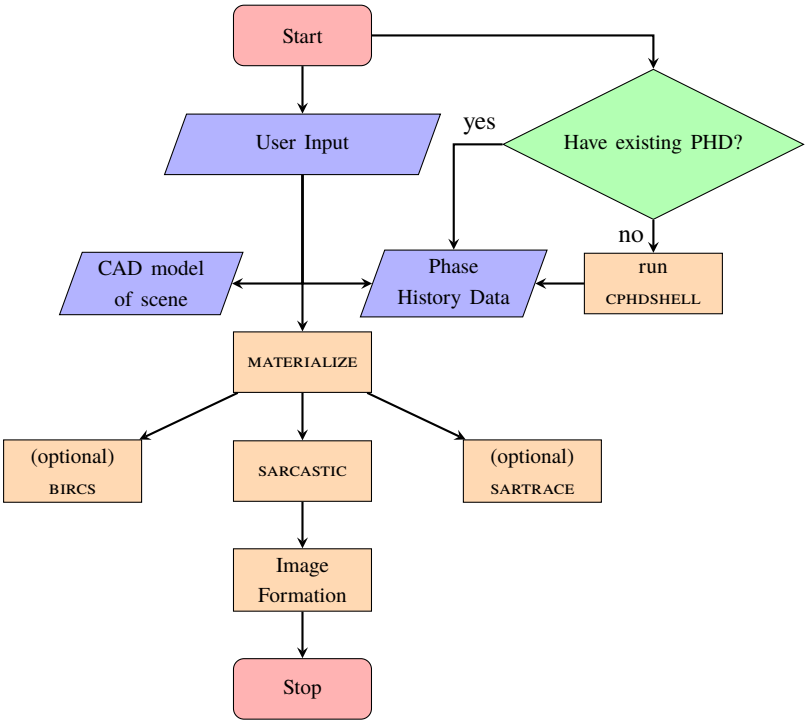


Figure 5.1: Process Flow of a Typical Run of SARCASTIC

was to use an existing SAR collection of a target area together with a CAD model of the analyst's scene. SARCASTIC would then use the CAD model and original collection data to simulate a comparison SAR image so that the analyst can gain a better understanding of the collected scene and, if required, adjust their CAD model to increase their understanding of the image. In many cases however, it is useful to be able to simulate a scene without a prior collection. In these situations, no prior collected data (and hence no common phase history data (CPHD) file) exists. If this is the situation then the package allows a CPHD *shell* to be constructed using user defined collection parameters and the CPHDSHELL program. This shell contains all the required information to process a SAR collection into a SAR image such as position and timing information (the narrowband data), but does not contain any of the received radar echo data (the wideband data).

Once the phase history data and CAD data have been validated, they are passed to the MATERIALISE function to convert the CAD triangle soup into a triangle mesh as described in Chapter 3. In addition to ensuring a Delaunay triangulated mesh for each type of material surface in the CAD model, it also applies a surface roughening

filter and electromagnetic resistive properties based upon the material type used in the CAD file.

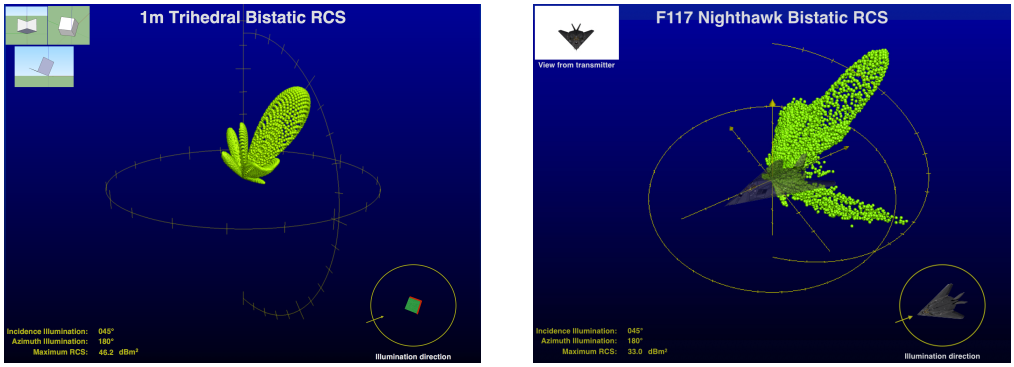


Figure 5.2: (Left) Bistatic Radar Cross Section of a $1m^2$ PEC trihedral illuminated into it's boresight at 10GHz. The maximum RCS is in the monostatic direction and is $44.2dBm^2$. (Right) An estimation of the Radar Cross Section of the F117-Nighthawk using a CAD model in the Sketchup® Warehouse

Following the triangle mesh construction, the user can then proceed with running SARCASTIC to replace or fill in the wideband data within the provided CPHD file. It is sometimes useful however to use the core internals of SARCASTIC for experimentation or diagnostics. Two additional programs have been developed for this:

- BIRCS - Bistatic Radar Cross Section tool. This program does not require an input CPHD file and instead uses user information to predict the radar cross section of the target. Rather than being SAR specific it is a simple radar RCS measuring tool. It uses the parallel processing engine within SARCASTIC however to rapidly calculate the radar cross section from all positions around a target (hence bistatic rather than monostatic) when illuminated from a user provided location and with a specified radar centre frequency. Figure 5.2 shows the bistatic radar cross section of a $1m^2$ PEC trihedral and also a predicted radar cross section of the Lockheed Martin F117- Nighthawk stealth aircraft taken from a CAD model publicly available on the Sketchup® Warehouse.

- SARTRACE. Sometimes it is useful to trace the path of electromagnetic rays through the scene. This is the function of SARTRACE. It allows the user to

select a pulse within a CPHD file (or location of a pulse from user input) and then provides a graphic depiction of the reflections and interactions. It is useful to understand multipath signatures in SAR imagery and identify the source of the signature's multiple reflections.

It is more usual to call SARCASTIC as part of a user's workflow. Algorithm 21 provides the SARCASTIC core algorithm.

Algorithm 21 CPHD wideband data construction using SARCASTIC

```
 1: procedure SARCASTIC
 2:     get user input
 3:     read in triangle mesh and CPHD file
 4:     if static scene then tree ← FASTKDTREE(M)                    ▷ on GPU
 5:     while pulses to process do                                   ▷ in parallel
 6:         generate field of rays from transmitter location
 7:         if not static scene then
 8:             apply motion translation to mesh M for this pulse time
 9:             tree ← FASTKDTREE(M)
10:         for all rays do
11:             while num of rays>0 and num of bounces < maxbounces do
12:                 test ray for intersection with scene
13:                 if intersection found then
14:                     generate reflection ray
15:                     generate shadow ray   ▷ ray from reflection point to receiver
16:                     if shadow ray does not intersect scene then
17:                         store RCS with total path length
18:                     build new rays from reflection rays
19:                     increment bounce count
20:         for all RCS range pairs do
21:             complex addition of signal into pulse receive window
22:     RETURN(pulses)
```

In this algorithm, the kd-tree is generated using a GPU whilst the iteration through pulses occurs as separate *posix threads* on the CPU. The reason being that the potential for many branch operations within the pulse loop mean that the GPU *warps* can quickly lose coherence and become significantly slower than the CPU implementation. This is primarily caused by the ray-triangle intersection calculation. Ideally, the GPU warp would be fully populated with single instruction multiple data (SIMD) instructions for this calculation but as the rays largely pass through different

paths within the scene, they undergo a different number of reflections and so higher bounce orders have wildly different intersection calculations.

There is however a simple optimisation that significantly improves performance which will now be discussed.

5.2 Optimisation Using Pulse Under-Sampling

As discussed in Chapter 1, there are strict requirements on the pulse repetition frequency (PRF) of a SAR sensor that are driven by the size of the illuminated footprint on the ground. Most simulated scenes however are significantly smaller than the illuminated area and so it is possible to *undersample* the azimuth bandwidth without consequence. This is achieved by only simulating every N pulses. To determine what N can be consider the requirement for the PRF when imaging a scene at broadside:

$$PRF_{beam} > 2v \sin\left(\frac{\theta_{beam}}{2}\right)\frac{f_c}{c} \tag{5.1}$$

with v being the platform velocity, f_c the transmit centre frequency and θ_{beam} the angular width of the half power transmit beam given by:

$$\theta_{beam} = \frac{\lambda}{d} \tag{5.2}$$

with d being the size of the antenna in the along track dimension and λ the centre wavelength of the transmitted pulse. If the simulated scene is considerably smaller than the scene size that the sensor is designed to illuminate then the bandwidth of Doppler frequency requiring adequate sampling is reduced and becomes:

$$PRF_{scene} > 2v \sin\left(\frac{\theta_{scene}}{2}\right)\frac{f_c}{c} \tag{5.3}$$

As a result, the collected data set can be simulated at a reduced pulse repetition frequency. The reduction factor is given by:

$$N = \frac{prf_{beam}}{prf_{scene}} = \frac{2v \sin\left(\frac{\theta_{beam}}{2}\right)\frac{f_c}{c}}{2v \sin\left(\frac{\theta_{scene}}{2}\right)\frac{f_c}{c}} = \frac{\sin\left(\frac{\theta_{beam}}{2}\right)}{\sin\left(\frac{\theta_{scene}}{2}\right)} \tag{5.4}$$

or, in terms of illuminated areas on the ground:

$$N = \sin\left(\frac{x_{spotlight}}{2R}\right) \Big/ \sin\left(\frac{x_{scene}}{2R}\right) \tag{5.5}$$

For a typical SAR collection, designed to illuminate a 1.2km scene from 20km stand-off range, a simulated scene of 100m can yield a reduction in PRF requirement by a factor of $N = 11$. This can be implemented by processing only every N pulses within SARCASTIC and yields a significant (factor of 11 in this example) performance increase.

5.3 Results

To validate the focusing[1] of SARCASTIC a series of simulated scenes were created and SARCASTIC used to generate the phase history data. The phase history data was then passed through a SAR image formation processor in exactly the same way as a collected SAR dataset and the resulting image compared.

Figure 5.3 presents a SAR image of a trihedral farm simulated by SARCASTIC. The collection parameters were from a hypothetical X-band SAR sensor imaging the array from low Earth-orbit and the phase history data used as input to the simulation was generated using the CPHDSHELL routine from Figure 5.1. This provides a useful way to debug the process as the input and output parameters were stable and known throughout the process. Using this approach, any phase errors introduced by SARCASTIC can be identified as they would contribute to either range or azimuth defocus within the final image. collected resolution of the input phase history data was set to 0.2m in both range and azimuth and the measured mean azimuth resolution after image formation was 0.21m and 0.20m in range suggesting that SARCASTIC is correctly simulating the raw data. The process also allows the multipath interaction of the ray tracing to be evaluated as without this the trihedrals would not appear as point targets requiring three bounces from the faces of the target. The mean radar cross section of the array of trihedrals was 43.8dBm^2. This is slightly lower than the expected value of 44.2dBm^2 for a 1m square trihedral at 10GHz. It is likely that the

[1]The RCS of the physical optics calculations were validated in Chapter 2

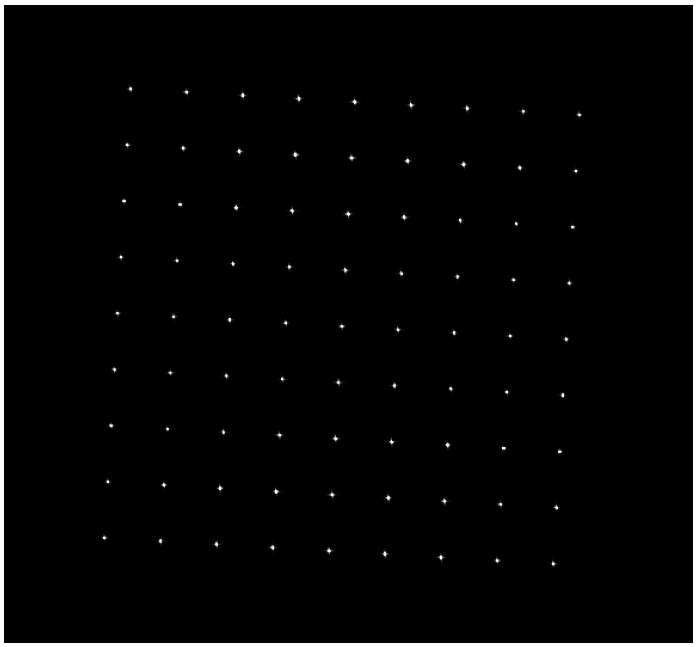


Figure 5.3: A simulated scene of 81, 1m square trihedrals. Mean RCS across the farm was 43.8dBm^2.

difference is due to the triangle density used in the MATERIALISE process. If too few facets are used then secondary and tertiary near-field reflections can have a slight error on the triangle's reflected signature. Although not large, these can add up and, in the case of a trihedral, reduce the coherent integration of all rays reflecting from the target. This can be addressed by reducing the facet size on the MATERIALISE step or by using a different EM field model such as the *method of moments* technique. Both approaches would increase the run time however.

Another way to validate the simulator is to compare a simulated image to a real SAR image collected from an airborne (or space-borne) sensor. For this an experimental Ku-band SAR sensor was used. The sensor is the Lynx SAR built and developed by Sandia National Laboratories (SNL) in New Mexico and was selected as it provides its collected data in a Common Phase History Data (CPHD) format. Experimental data of Kirtland Air Force Base in New Mexico was provided of a selection of runs and included aircraft and main battle tank (MBT) targets. One of the targets was a T72 Russian Tank and a CAD model of the vehicle was obtained from the Sketchup Warehouse® [98] which can be seen in Figure 5.4

The model was adjusted to assign properties to the vehicle and situate it on

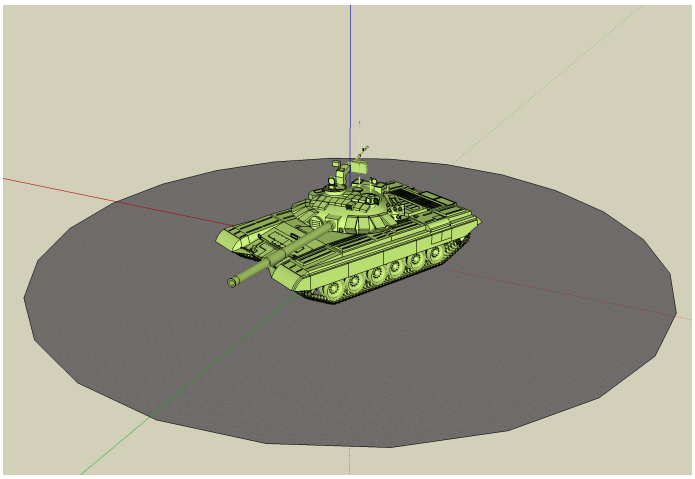


Figure 5.4: CAD model of a Russian T-72 Main Battle Tank taken from the Sketchup Warehouse® [98]

a concrete turntable surrounded by short grass. This was then passed through MATERIALISE to Delaunay triangulate the mesh and apply surface roughness and EM properties to the materials. The results can be seen in Figure 5.5. The MATERIALISE process resulted in de-constructing the scene into 7 million planar facets.

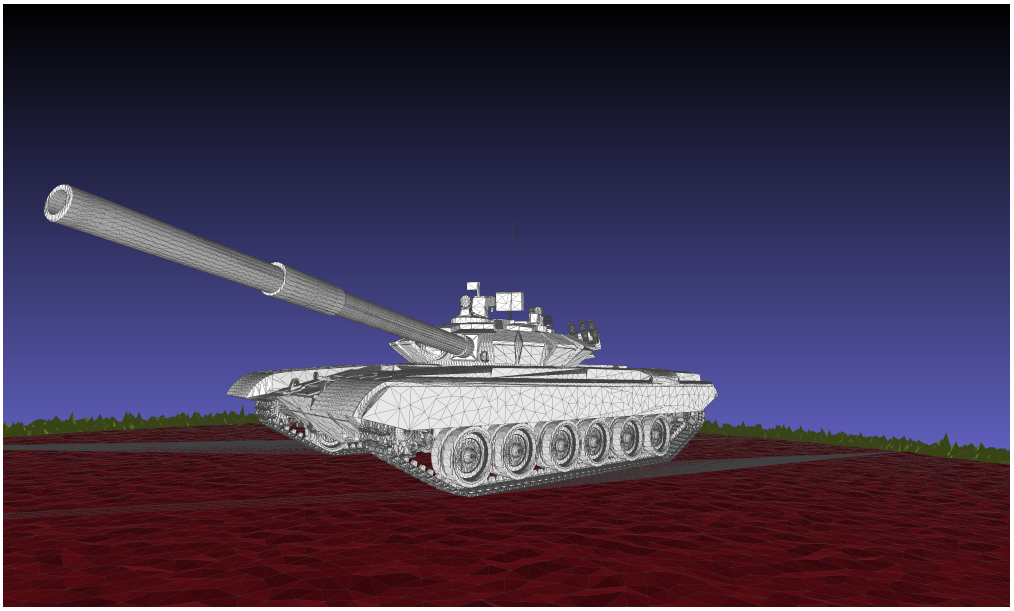


Figure 5.5: CAD model of T-72 Russian Tank after being passed through the MATERIALISE algorithm. Note that the facets are now Delaunay and that surface roughness has been applied to the concrete apron(red) and vegetation. This model contains 7 million facets

The next step is to simulate the phase history data of the radar collection using

SARCASTIC. The actual collection was obtained at a resolution of 0.1m in range and azimuth. The SARCASTIC process took 191 minutes to run in this example on a Late 2013 24 core Apple Mac-Pro with 64GB RAM.

The final stage is to pass both the original phase history data and the simulated phase history through the same SAR processor and compare the results. These can be see in Figure 5.6.

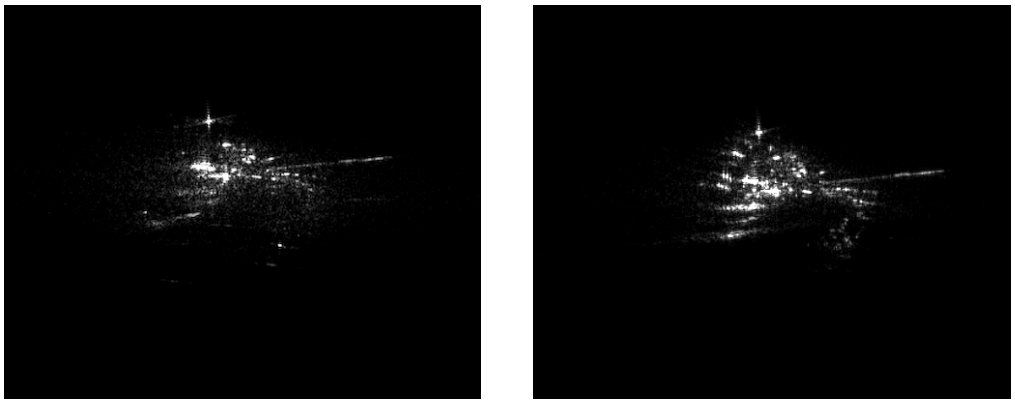


Figure 5.6: (Left) SAR image of a T-72 MBT collected by the SNL Lynx SAR system. Resolution in 0.1m (*Courtesy Sandia National Laboratories*). (Right) Simulated SAR image of a T-72 MBT taken from the Sketchup Warehouse®.

It is encouraging to see that both the simulated and actual imagery have remarkable similarities. Especially as the position of the gun barrel and orientation relative to the sensor had to be estimated. This result was achieved with a significant amount of trial and error to ensure consistent alignment. This could only be achieved in a realistic timescale with a simulator that allowed flexibility and timeliness of repeated runs and is a key requirement for SARCASTIC.

Perhaps more interesting are the areas that are not consistent between the two images. The true SAR image has many more diffuse interactions most likely caused by the surfaces not being as planar as the simulated model. This may be due to wear and tear or other environmental factors. Equally there are likely differences in the estimated electromagnetic properties of the vehicle in addition to the surface roughness of the ground plane. It is clear that whilst the general shape and size of the simulated target is a close match to the true image, subtle imperfections in the model caused significant differences on the vehicle body.

5.4 Recommendations for Improvement

Although the results from the previous section are encouraging and likely to prove useful in the analysis of SAR imagery, there are several areas where the suite of tools can be improved. Perhaps the most obvious one is in the core SARCASTIC algorithm which although being multi-core with GPU elements, does not fully exploit the efficiencies of SIMD processing throughout. One way to potentially improve this would be to adapt Algorithm 21 to process one ray traversal at a time for all pulses. As this is more likely to be a SIMD process it lends itself better to GPU processing. After each traversal, the relevant parameters could then be stored to process the RCS for all pulses again on the GPU. An approach such as this would almost certainly be faster but would be challenging to implement on a moving scene without storing a different kd-tree for each pulse.

Another obvious improvement would be the use of a more accurate EM field calculator. Currently SARCASTIC uses *physical optics* which has been shown to be accurate enough for SAR analysis where shape is more significant that absolute RCS. However for more sophisticated analysis, such as imaging targets comprised of metamaterials or even targets with antennas or electronic circuits, a better approach would be to use Finite-DIfference Time-Domain techniques [84] which lend themselves well to GPU processing.

Finally, if absolute RCS is important, SARCASTIC only makes assumptions on the actual electromagnetic properties of materials. A useful activity would be to compare RCS values predicted by SARCASTIC with measured RCS values for large areas of homogeneous material, (such as concrete, wood, asphalt etc). This would provide useful tables that could be used for comparison of one material over another and potentially offer the capability of identifying a material purely form its orientation, shape and RCS.

5.5 Summary

This chapter builds on the background material on EM theory (Chapter 2) which, when combined with the ray-tracing technology presented in Chapters 3 and 4

facilitates the development of a ray-tracing simulator. The Chapter details the implementation of this simulator which notably produces phase history data as it's output so that the data can be passed through the same SAR image formation processor as a comparative image. This allows a user to examine the scene with the same processor-induced artefacts as the actual SAR collection and precisely aligns the simulated data with the true data.

The chapter also presented a way to reduce the simulation run time by *pulse under-sampling*. It was shows that this approach can reduce the simulation time by an order of magnitude, depending on the ratio between simulated scene size and actual imaged area. Of interest, this is similar to a SAR image formation approach commonly used when processing small regions of a SAR image that are part of a larger SAR scene E.g. [152].

Finally a simulated SAR image was formed for comparison with a true SAR image of a T-72 main battle tank. The section concluded with recommendations fr improvement.

The next two chapters will look at two particular use cases for SARCASTIC, multipath and vibration analysis.

Chapter 6

Multipath Analysis

A key challenge associated with SAR image understanding is the identification and interpretation of radar *multipath*. This is a feature of SAR imagery where secondary and higher order reflections are focused to provide a downrange replica of a feature within the scene. This is a task ideally suited to a ray tracing simulator and will be covered in this chapter.

Consider Figure 6.1 which shows a simplified situation to demonstrate multipath phenomenology. In this situation a void or *silo* is located slightly nearer to the radar than a step. The step forms a natural dihedral reflection and all the energy incident on the face of the step or the ground in front of it has the same round trip path length at the radar receiver. As a result the energy integrates coherently and so a bright feature is formed.

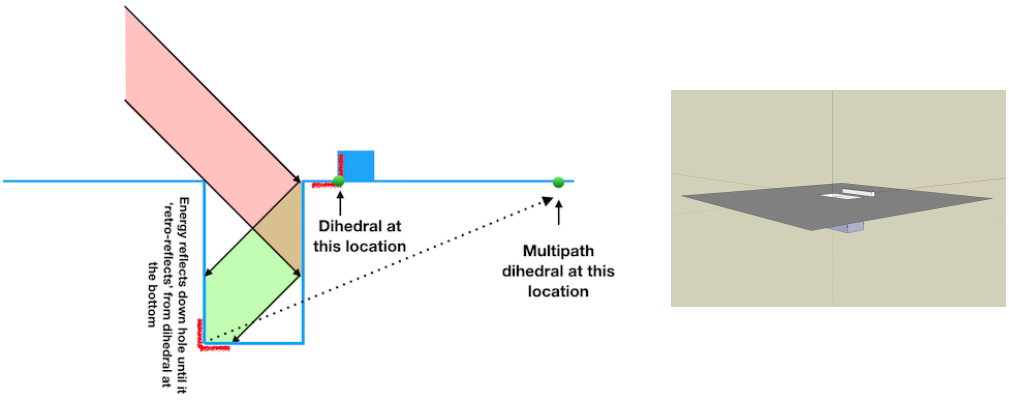


Figure 6.1: (Left) Illustration of a radar void or Silo and the radar path trajectories within it. (Right) a CAD model of the silo scenario prior to simulation with SARCASTIC

The energy that penetrates into the silo however undergoes multiple reflections

before it is received back at the radar. The total round trip path length back at the receiver is therefore longer than that of the vertex at the step and so the multipath radar returns from the silo appear further down range in the output SAR image. This effect can be seen in Figure 6.2 which shows the resulting SAR image after the CAD model in Figure 6.1 has been simulated with SARCASTIC.

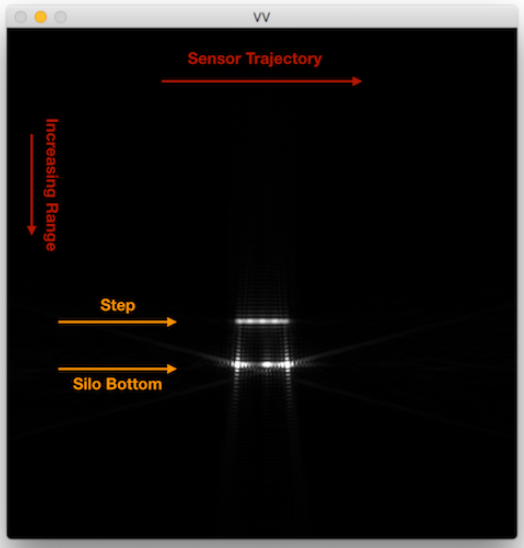


Figure 6.2: SARCASTIC SAR image of the silo scenario (VV polarisation). The direct return dihedral is nearer in range than the multipath signature form the bottom of the silo.

It is interesting to note that in Figure 6.2 the multipath signature has bright points at both the left and right side. This is due to the synthetic aperture being formed over a wide range of angles (10° in this case). At the start and end of the synthetic aperture, the bottom corners of the silo form a trihedral whereas the centre of the silo remains a dihedral that is only illuminated from a broadside position during the central region of the synthetic aperture.

6.1 Polarisation

It can be quite challenging to differentiate a direct return from a multipath return, especially from an open void such as Figure 6.2. A common way to identify and isolate multipath effects is to use polarisation if it is available as it allows a user the

opportunity to perform a level of advanced exploitation. The first step is to recognise that the total scattering of a resolution cell measured by the radar $[S]$ is made from a combination of scattering objects within the resolution cell given by:

$$[S] = \sum_{i=0}^{k} w_i [S]_i \tag{6.1}$$

where $[S]_i$ is the scattering matrix for the ith object within the cell. w_i is the weighting applied to each object. Usually there are many scattering objects within the cell that coherently combine to provide a distribution of phase and polarimetric responses. With certain types of target and particularly at finer resolutions, some of the scatterers can appear canonical and dominate the resolution cell. In these cases it is possible to determine the nature of the canonical scatterer using a polarimetric decomposition. Many polarimetric decompositions exist[1] and have been considered previously for multi-path analysis [154]. A detailed analysis is therefore not required here but an overview of the simplest decomposition is useful for comparison later.

6.1.1 Pauli Decomposition

The Pauli Decomposition splits the scattering matrix $[S]$ into four basis derived from the original VV, VH, HV and HH linear polarisations. The basis correspond to :

$$[S]_a = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{6.2}$$

$$[S]_b = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \tag{6.3}$$

$$[S]_c = \frac{1}{\sqrt{2}} \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \tag{6.4}$$

and

$$[S]_d = \frac{1}{\sqrt{2}} \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \tag{6.5}$$

[1] [153] provides a useful tutorial

As reciprocity applies in a monostatic configuration, $S_{vh} = S_{hv}$. In which case the total scattering matrix in terms of linear polarisations can be written as

$$[S] = \begin{bmatrix} S_{hh} & S_{hv} \\ S_{hv} & S_{vv} \end{bmatrix} \tag{6.6}$$

which can be decomposed into :

$$\begin{bmatrix} S_{hh} & S_{hv} \\ S_{hv} & S_{vv} \end{bmatrix} = \alpha[S]_a + \beta[S]_b + \gamma[S]_c \tag{6.7}$$

with

$$\alpha = \frac{S_{hh} + S_{vv}}{\sqrt{2}} \qquad \beta = \frac{S_{hh} - S_{vv}}{\sqrt{2}} \qquad \gamma = \sqrt{2}S_{hv} \tag{6.8}$$

This provides a useful way to represent the full scattering matrix by assigning colours to the different values of α, β and γ.

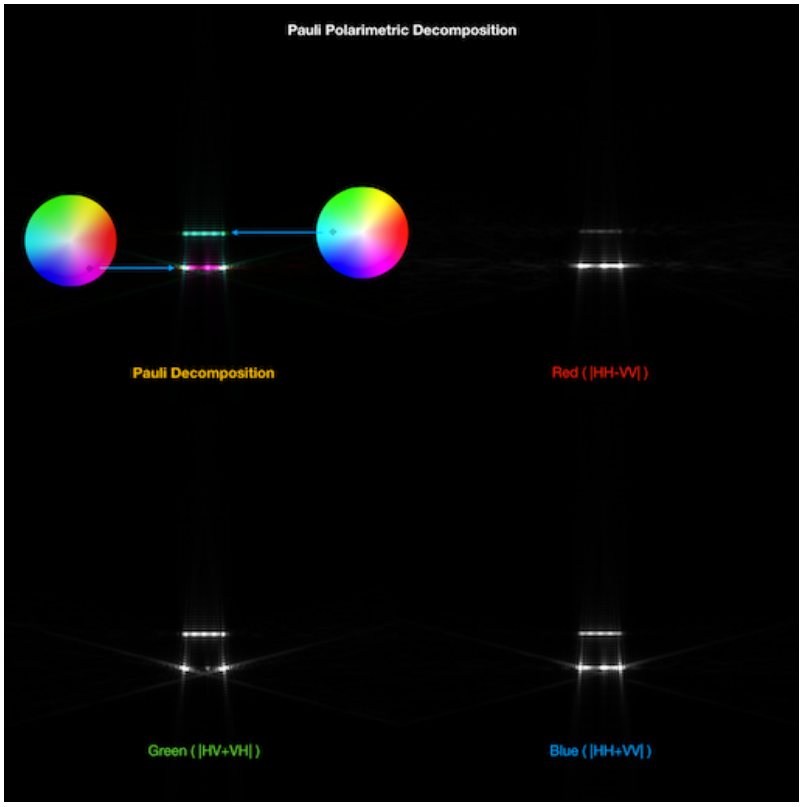


Figure 6.3: Pauli Decomposition of the silo CAD model simulated with SARCASTIC with α = red, β = blue and γ = green.

The interpretation of the Pauli Decomposition requires a general statement to be made regarding the meaning of the α, β and γ parameters in 6.8 when associated with canonical scatterers. In this case $[S]_a$ corresponds to scattering types that are strong in their co-polar returns HH and VV. This is typical of single or odd-bounce canonical scatterers such as flat surfaces, spheres or trihedrals. The $[S]_b$ scattering matrix corresponds to those canonical scatterers that have a significant difference between their co-polar horizontally polarised scattering (HH) and their co-polar vertically polarised scattering (VV). This usually corresponds to dihedral scatterers. By extension this also applies to even bounce scattering. The third $[S]_c$ scattering matrix relates to those scattering mechanisms that translate a V polarisation into a H pol or vice versa. A canonical example would be a dihedral reflector aligned with its seam at 45° but a more natural and common example is the diffuse scattering from vegetation.

A common representation of the Pauli Decomposition assigns blue to $[S]_a$, green to $[S]_c$ and red to $[S]_b$ [153]. An example can be seen in Figure 6.3 which shows a Pauli Decomposition of the four polarisation images simulated with SARCASTIC of the silo. It is worth stating the such a colour representation is only indicative of the types of scattering mechanisms as each of the colour channels have to be individually normalised in order to prevent any single more dominant channel (usually $[S]_a$) from dominating.

The top left image in Figure 6.3 shows the different polarimetric responses caused by the additional multiple bounces down the silo. The lower pink signature is a mixture of both red ($|HH - VV|$) and blue ($|HH + VV|$) which indicates that the signature has strong returns in all channels but significantly has a larger response in one of the polarisations (H here) than the other.

The higher, cyan coloured signature is caused by bright returns in all channels but particularly in the area of cross polarised (diffuse) scattering. For comparison the original polarimetric channels are presented in Figure 6.4

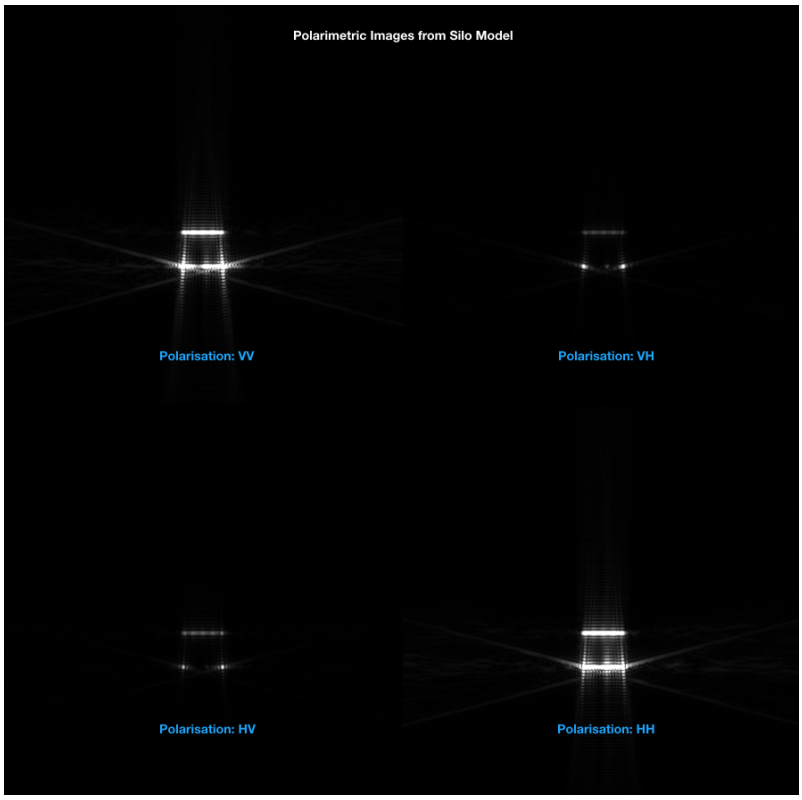


Figure 6.4: Original polarisation channels corresponding to Figure 6.3 prior to Pauli Decomposition

6.2 Ray Unravelling

Whilst polarisation can provide useful insight into the nature of a scatterer, interpretation can be difficult, especially if the scatterer is not canonical. Also not all SAR sensors have a fully polarimetric collection capability. A different approach is presented here - *ray unravelling*.

This approach could be considered to be hypothesis based and relies on the user being able to simulate a SAR image as a suitable representation of the actual collected SAR data. The rapid reprocessing time of SARCASTIC allows a user to quickly adjust a CAD model and reprocess the scene until it represents the target area, during this process the inspection of the CAD model provides a level of spatial understanding of the primitives within the scene that gave rise to the scattering features within the true image. On some occasions, especially where multipath is concerned where the number of reflections is not obvious, a further level of interrogation is required.

The analysis proceeds as in the flowchart 6.5

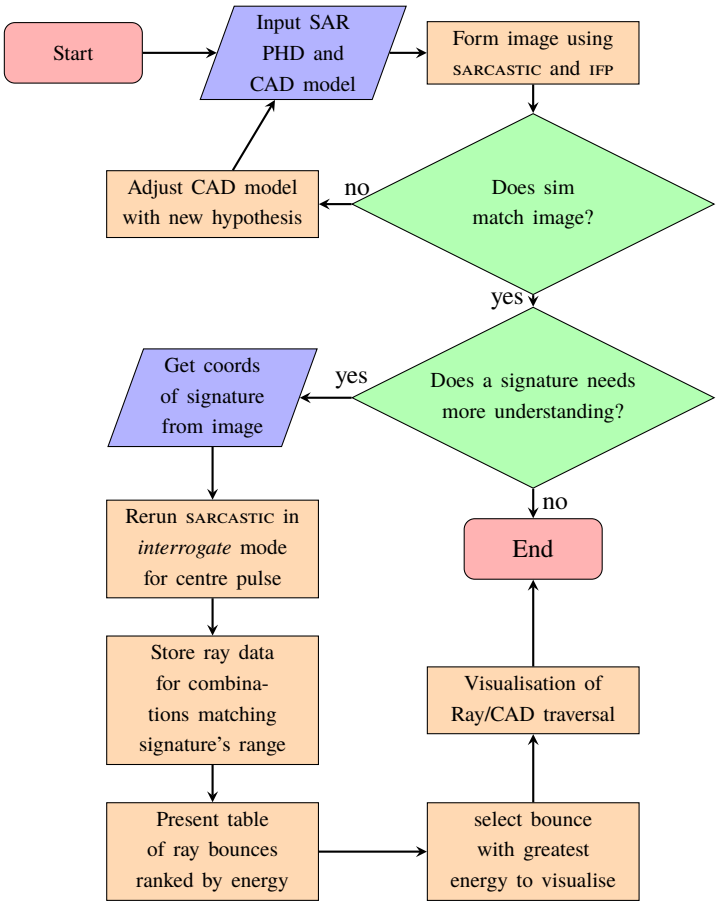


Figure 6.5: Process Flow to Ray-unravel a Signature Within a SARCASTIC Image

The concept is relatively simple. Once a reasonably accurate representation of a SAR image has been formed using SARCASTIC any unexplained artifacts can be interrogated in a more forensic manner by looking at how they were formed within the collection. A second ray-casting is used with the same parameters as the original simulation but this time with the specific purpose of storing any interactions that have the same total round trip path length as the selected pixel being interrogated within the SAR image.

To demonstrate this approach the multipath signature in Figure 6.3 was selected for an interrogation run. In this case the brightest energy return for the signature corresponded to bounce order three (The fourth bounce as the first is bounce order 0). The ray metadata corresponding to four bounces at the precise range of the signature within the SAR image was then used to unravel the ray traversals within the kd-tree. Figure 6.6 shows the ray tracing paths (in green) for the brightest responses.

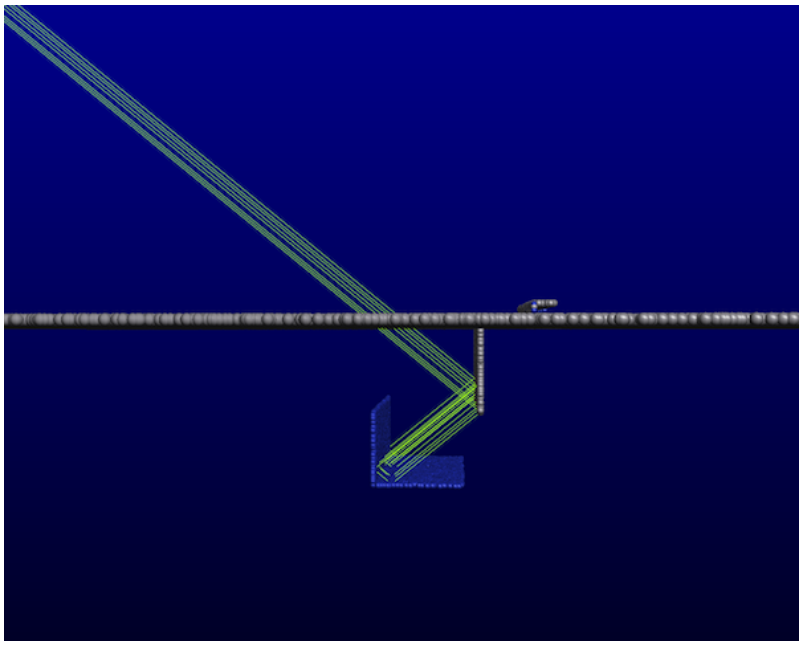


Figure 6.6: The pink multipath signature from Figure 6.3 was interrogated and shown to correspond primarily to a three bounce interaction. This was then ray-unravelled. The figure shows the silo from the side with the ray traversal steps into the silo as green lines.

6.3 Application to Real Data

To validate the approach it was decided to use SAR data from a Defence Science and Technology Laboratory (DSTL) trial called 'Hydravision II' that took place in May 2010 at Dstl's headquarters at Porton Down. The aim of the trial was to look at polarimetric decomposition techniques using an X-Band fully polarimetric system called F-SAR provided by DLR. Table 6.1 provides a summary of the F-SAR collection system.

Figure 6.7 provides an overview of the site together with a typical SAR image from the trial. Of particular interest is the firestation within the scene (highlighted) that was observed with different door configurations on different days of the trial. This was selected as a suitable target with the specific intention of seeing if SARCASTIC's ray-unravelling capability can provide insight into the contents of the interior of the building. Unfortunately the phase history data for the trial was not available and so a suitable *shell* CPHD file was created using the collection parameters within the header files of the trial. Whilst this provides a suitable approximation to the

Parameter	Value
Sensor	F-SAR (DLR)
Date of collect	12th May 2010
Tasked SRP	$51.132244°N, -1.694279°E 155.40m$
Collection mode	SPOTLIGHT
Collection geometry	Monostatic
Centre grazing angle is	45.0°
System PRF	4032.25 Hz
Channel PRF (quad pol)	1008.06 Hz
Transmit bandwidth	760 MHz
Slant range resolution	0.197 m
Transmit centre frequency	9.6 GHz
Velocity at aperture centre	86.6 m/s
Azimuth resolution	0.5 m

Table 6.1: Nominal F-SAR Parameters for Hydravision II Trial

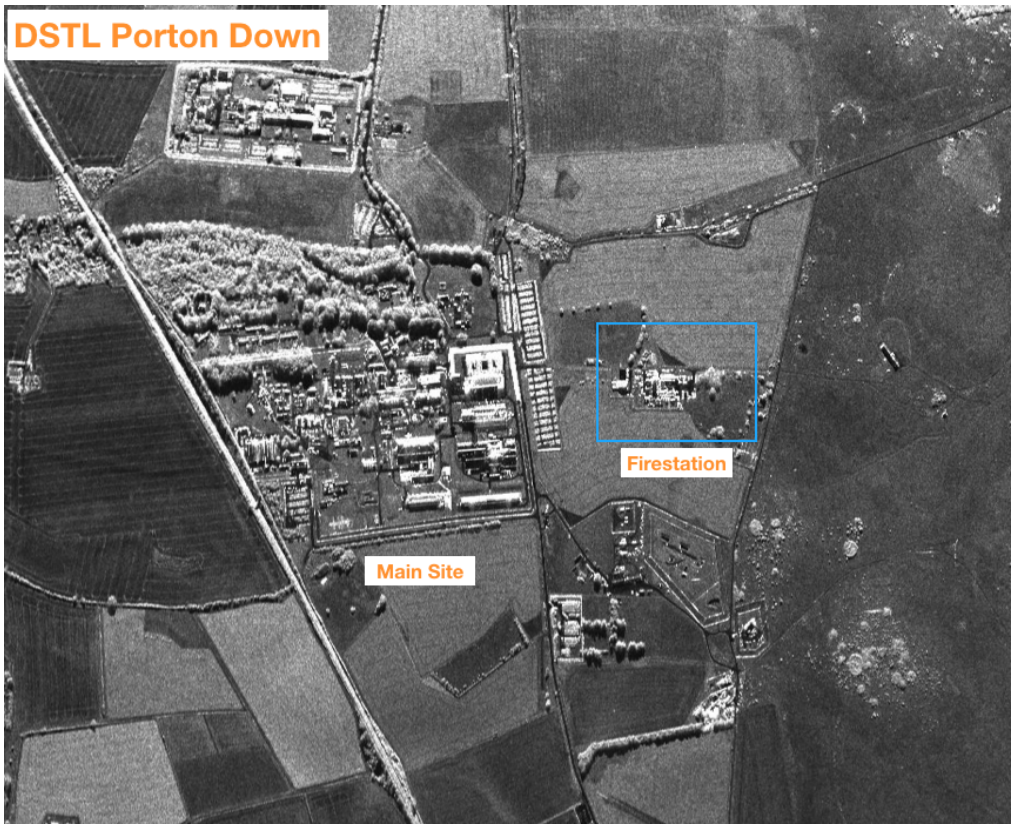


Figure 6.7: FSAR X-band SAR Image of Dstl Porton Down

collection geometry of the system it does not enable a precise comparison between SARCASTIC and F-SAR as the location for each pulse within the actual SAR collection is unknown.

To start with, a CAD model of the firestation was developed and can be seen in Figure 6.8 together with a selection of ground truth images for comparison. It is worth noting that there is a training tower outside the firestation which was not included in the CAD model as it increases complexity whilst not adding to the analysis. A close up view of the firestation in the F-SAR imagery can be seen in Figure 6.9 together with a Pauli polarimetric decomposition (inset) built from the four polarisations after SARCASTIC simulation of the CAD model.

Figure 6.8: CAD model and ground truth images of the firestation at Porton Down. Note that the training tower (top right) was not built into the CAD model.

The CAD model used to make the polarimetric decomposition in Figure 6.9 was built with the left hand door of the three doors open so that the multipath interactions within the building could be observed in terms of their polarimetric signatures. The

F-SAR collection in the image has all three doors open and shows faint multipath signatures appearing to extend beyond the far side of the building.

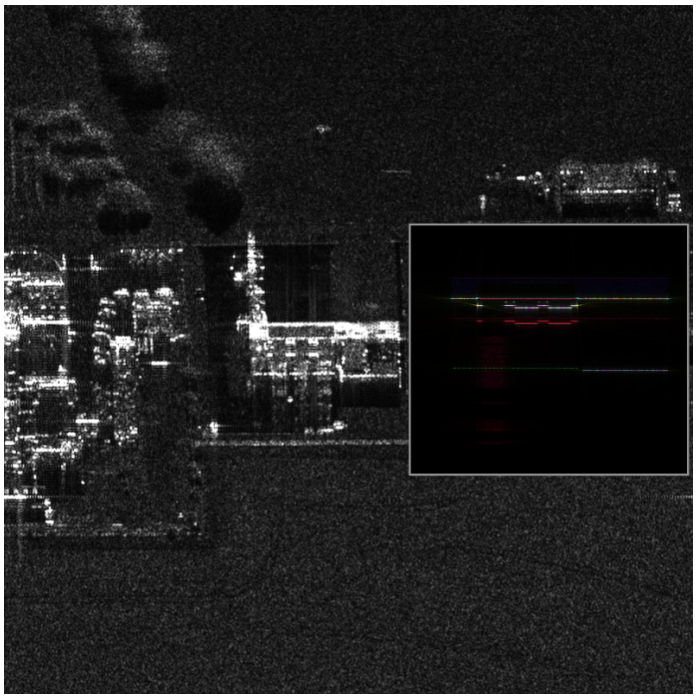


Figure 6.9: F-SAR Image (HH polarisation) taken of the firestation at DSTL Porton Down with (inset) a Pauli Polarimetric decomposition of the CAD model of the scene using SARCASTIC

The polarimetric signatures provide an initial indication of the types of scattering modes occurring within the building. It can be seen that the front doors of the building have two signatures one white and one red. The red signature indicates a strong polarimetric component (see Figure 6.3) associated with a reflection from the under side of the top of the door. This compares favorably with the analysis performed in [154]. In a similar way, a faint red multipath signature can be observed just below the left hand open door, again implying a strong polarimetric response from the underside of the roof inside the building. The far range end of the roof exterior has a bright reflection shown as a green line in the SAR image corresponding to having a cross-polarimetric component, in this case caused by the polarimetric scattering of the dihedral as a function of aperture position.

To gain an understanding of the issues associated with understanding the building interior, the CAD model was adjusted such that a fire engine was present in the

central bay. It was also observed that ground truth imagery (Figure 6.10) showed the the left hand bay had several shelves present in it during trial and so the doors were opened for this bay in the CAD model. The scene was then simulated again and the results can be seen in Figure 6.11.

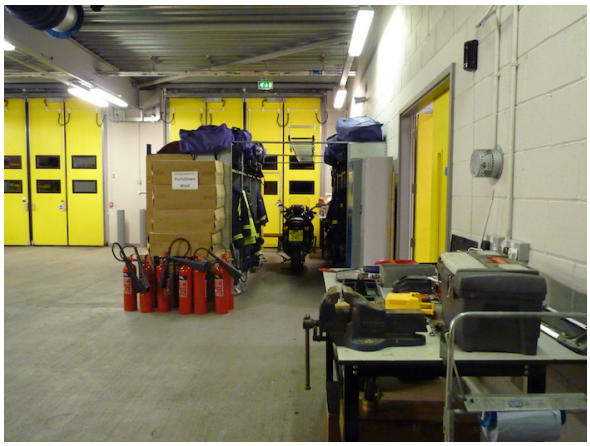


Figure 6.10: Handheld ground truth image of left bay within the firestation showing that it was not empty

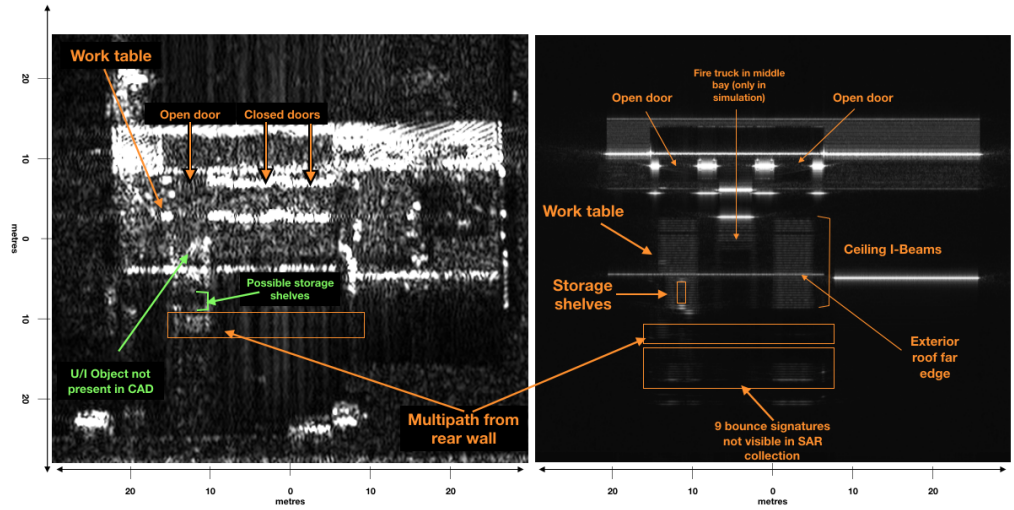


Figure 6.11: Comparison of SARCASTIC simulation of firestation (right) with F-SAR imagery (left). (Gray scale represents amplitude and has been stretched to highlight key features)

Upon first analysis the simulated image compares remarkably well with the F-SAR image with the direct scattering features from the exterior of the building aligning with those signatures within the F-SAR image. This F-SAR image was

collected early in the Hydravision trial with only the left hand door open whereas the CAD model had all three doors open. This can clearly be seen in Figure 6.11.

Some of the other signatures within the interior of the firestation are more challenging to explain. To overcome this the ray-unravelling process described above was used to identify features within the simulated image. If these signatures have a corresponding signatures within the true SAR image then the level of confidence of the source of the signature can be raised. Figure 6.11 has been marked with key features identified using this process. There are however two features, highlighted in green, were, the source of the signatures is not as obvious. The first is the feature marked as unidentified ('U/I') object. This is quite bright within the F-SAR image and was not represented in the CAD model hence it does not appear in the simulated scene. This may be the presence of the fire extinguishers or other small, metal objects near to the ground. Another possibility is that it could be the motorcycle seen in Figure 6.10. Ultimately it is impossible to know for certain. The second uncertain signature is interesting as it obscures the multipath signature from the underside of the roof in the exact same region as the shelves in the CAD model. The CAD model objects however have brighter start and end signatures caused by dihedral returns between the base of the shelves and multiple bounces between the top of the shelves and the underside of the roof. It is possible that the reduction in brightness of these signatures is caused by the presence of the several irregular items on the top of the shelves and in front of them (seen in the hand held imagery). The ray path generated through ray-unravelling can be seen in Figure 6.12. It is possible therefore that the slight shadow area in the true SAR image is caused by the shelves. Although the source can not be identified for certain, the lack of multipath in this region does show that the the bay is not empty.

6.4 Summary

This chapter has presented a discussion on the application of ray-tracing simulations to multipath analysis within SAR images. The understanding of multipath signatures is an essential skill for a SAR image analyst, made particularly challenging as there

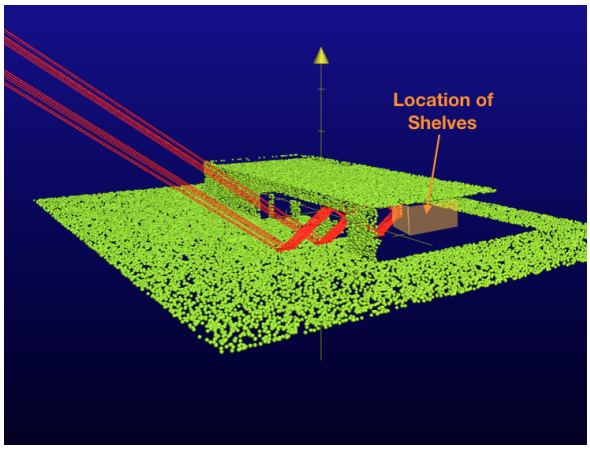


Figure 6.12: Ray-tracing image of the firestation showing the path of rays that create the front of the 'shelf' signature in the SARCASTIC scene in Figure 6.11

is no traditional way of knowing if a bright signature is a direct observation or a multipath artifact. This problem is conventionally overcome by an analyst applying contextual information to his assessment. This often requires additional collateral imagery (such as electro-optical images) together with a rigorous application of trigonometry to the precise collection geometry. As a result challenging targets like building interiors and cavities are only exploited by the most advanced and patient analysts.

The chapter discussed the application of previously well-used polarimetric decomposition techniques. Having four polarisations and a suitable polarimetric decomposition (here the Pauli decomposition was applied) provides a useful way to colour different types of multipath signatures. The application of polarimetric techniques has its own challenges. First the collecting radar must be able to transmit and receive with both vertical and horizontal polarisations. This increases the pulses repetition frequency by a factor of two and the volume of data by a factor of four. In most operational cases, data rate is a rate-determining quantity and it is often preferable to double the resolution in both range and azimuth directions rather than have four polarisations. Should a fully polarimetric SAR sensor be available however, the application and understanding of the applied decomposition can be just as challenging as the understanding of the original SAR images and there are currently no rigorous standards applied to colour or type of decomposition used.

Experienced users are rarely troubled by this but equally experienced users are very capable of applying a geometric analysis to the SAR image.

A technique called ray-unravelling was then presented as a by-product of the SAR simulation process. This allows a user to interrogate a feature within a SAR image and then visualise the path that rays had to have travelled to form the signature. This is a conceptually simple process and allows an analyst to more easily understand the nature of complicated scattering signatures. The approach does have limitations however in that only signatures that are present in the simulated scene can be unravelled which in turn means that the original CAD model needs to be a reasonable representation of the scene being imaged and the user must be able to make sensible postulations about the source of signatures. Nevertheless, a fast ray-tracer and SAR simulator affords a user with the ability to apply *trial-and-error* techniques to their analysis.

The chapter concludes by comparing the approaches discussed with real SAR data collected over DSTL's headquarters at Porton Down. It was shown that both the polarisation and ray-unravelling can provide additional understanding of the nature of the scene. It was shown the the presence or absence of objects within a building can be deduced without having a direct observation of the buildings interior. Whilst it was impossible to say exactly what was the source of key building interior features, sensible guesses could be made due to their size and radar cross section, which, with enough time and consideration, and combined with multiple SAR collections from different angles, may allow an analyst to better postulate the hidden contents of a building.

Chapter 7

Motion And Vibration

7.1 Introduction

The ability to rapidly re-build a k-dimensional tree of a simulated CAD scene opens up the possibility of accurately simulating moving features in the phase history of a SAR collection as the tree can be rebuilt once for every pulse (or every few pulses). In addition to conventional static or moving targets, a useful artifact to study is the modulating target. The term *modulating* refers to two types of object:

- Vibrating objects where the spatial position of the target changes periodically during the SAR collection.
- Flashing objects where the radar cross section of the target changes periodically during the SAR collection.

Radar returns from targets that are modulating during a SAR collection produce distinctive *paired echo* signatures within the formed image (figure 7.1). Such signatures are often treated as unwanted artifacts although they can provide additional useful information on the nature of the target being imaged. Unfortunately these signatures are subtle and only visible in imagery from SAR sensors that have a very high signal to clutter ratio, most commonly from a fine resolution SAR system. In addition, papers detailing the nature of vibration signatures [45], [155], [46] rarely consider the final form of the signature and are usually provided from the perspective of higher frequency phase distortions rather than observed signature characterisation. A description of the limitations of high resolution SAR processing is now

provided as a context on which to describe the nature of modulating signatures as observed in SAR imagery. The form of the modulating signature is then described and the key parameters defining its structure are derived. SARCASTIC will then be used to simulate vibration signatures from a vibrating generator and compared with real data. Finally a technique to detect the vibrating signatures will be described and results on real data presented.

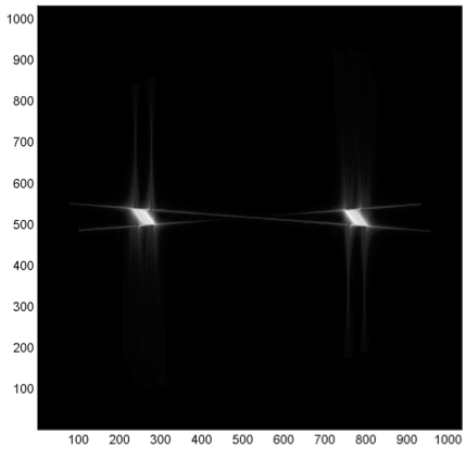


Figure 7.1: Simulated SAR image of a point target that has amplitude modulation whilst being imaged. The target's signature is represented by two displaced 'fuzzballs' located either side (azimuth direction) of the true target location.

7.2 Form of Modulating Signal

It is common to describe the synthetic aperture process in terms of a Doppler shift applied to returns from a transmitted pulse caused by the motion of the sensor. This line of thinking has an inherent problem when considering time-related effects such as modulations. A Doppler shift implies a change in frequency as a function of time but this is strictly incorrect for a SAR system which observes a varying rate of change of *phase* as a function of synthetic aperture angle. Unlike a static target, a modulating target changes its radar response as a function

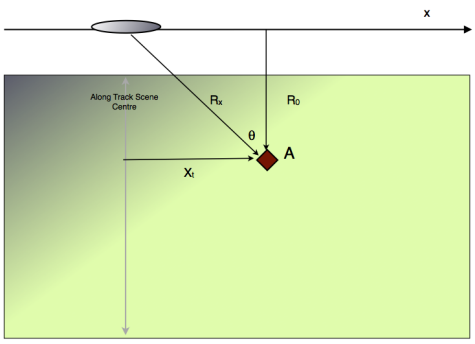


Figure 7.2: Simple Representation of SAR Collection Geometry

of 'true-time' whereas the perceived change

in frequency due to the Doppler effect of a static target is a function of the change of viewing geometry and would amount to the same change in phase irrespective of the velocity of the sensor (although the change in phase per unit time would change). It is therefore useful in this section to consider that a synthetic aperture is created through the measurement of the rate of change of range to a target as the sensor transits past it. Consider figure 7.2 in which the sensor illuminates a scene with a constant PRF and a constant platform velocity. The geometry of the scene is such that the range to the target as a function of along track position $R(x)$ is given by:

$$R(x) = \sqrt{R_0{}^2 + (x - X_t)^2} \tag{7.1}$$

where R_0 is the stand off range and X_t is the location of the target relative to the scene centre in the along track direction. The range and hence phase of the radar return is then stored as a 'phase history' which is later combined through the IFP to collapse the entire phase history into a single point. The form of (7.1) can be seen

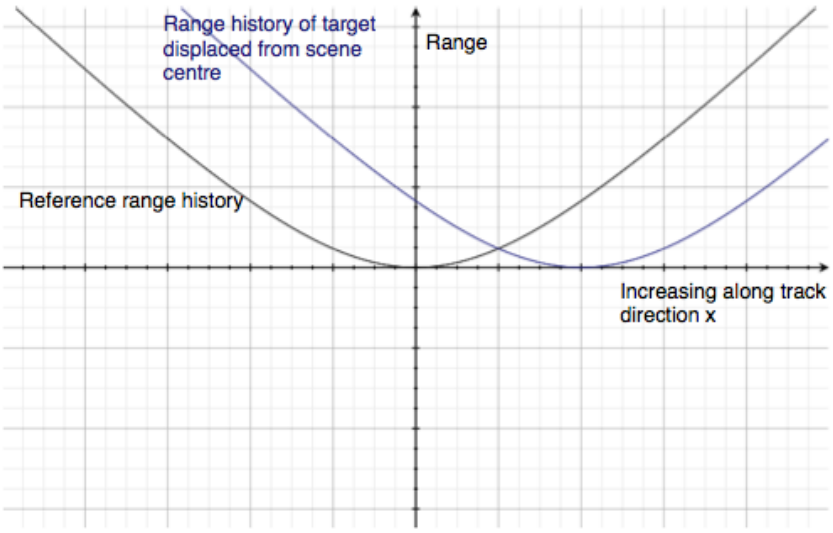


Figure 7.3: Target Range as a Function of Position Through Synthetic Aperture

in Figure 7.3. A key task in forming the synthetic aperture is to convolve the phase history of the azimuth data set (containing all samples for a given across track range) with the expected phase history for the target set. There is an issue however as the

form of equation (7.1) is nonlinear in x and, as x increases, $R(x)$ tends towards x. As the signal history is uniformly sampled in the x direction, the rate of change of range to a target between one pulse-sample and the next is therefore not constant over the synthetic aperture duration. (For the rate of change of range to be constant, $R(x)$ in equation (7.1) would be a function of x^2). The rate of change of range with along track sensor position can be derived from the first derivative of (7.1) to provide:

$$\frac{d}{dx}R(x) = x\left(R^2 + x^2\right)^{-\frac{1}{2}} \tag{7.2}$$

This is shown in Figure 7.4.

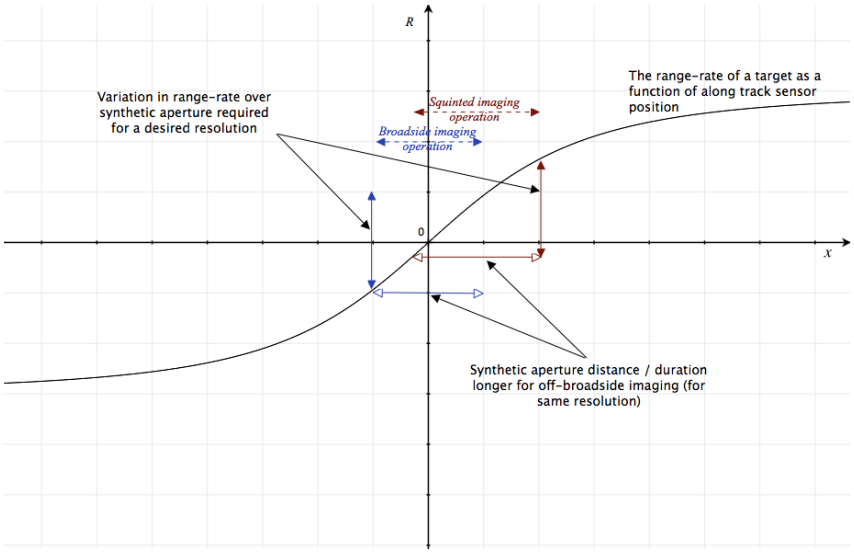


Figure 7.4: Pulse-to-Pulse Range Rate as a Function of Along Track Position

It can be seen that the change in range-rate when the sensor is a large distance before or after the broadside imaging position is significantly less than when the sensor is imaging from broadside. The pulse-to-pulse variation of range provides a change of sensor-to-target phase through the equation (7.3):

$$\phi(x) = \frac{4\pi}{\lambda_c}R(x) \tag{7.3}$$

Although this phase change varies with sensor position (x) and not actual time, it is often associated with a Doppler frequency[1]. The synthetic aperture is formed

[1]Frequency is defined as the *rate of change of phase* and so for SAR, the *rate of change* is more

by coherently summing returns from the subtended Doppler bandwidth and so the variation in range-rate over a synthetic aperture duration is the actual mechanism that provides along track resolution. It can therefore be seen from Figure 7.4 that for a pre-determined imaging resolution, and hence a required span of pulse-to-pulse range-rates, the sensor is required to stare for a longer synthetic aperture duration as the imaging operation moves away from broadside. It can also be seen in Figure 7.4 that the range-rate varies (approximately) linearly over the central region of the graph. This is the region described in the introduction to SAR (Chapter1) and is the usual situation for most SAR systems where it can be assumed that the phase history of a target's radar returns are quadratic in nature (i.e. the Doppler frequency, or rate of change of phase (or range) with sensor position, is a *linear*, frequency-modulated chirp waveform). Only when larger squint angles, low frequencies or very fine resolutions are required do the non-linear effects of Figure 7.4 become an issue. The effect of the non-linear form of Figure 7.4 on a target's impulse response can be seen in Figure 7.5. In this figure, the received signal from target *A* (from Figure 7.2) has

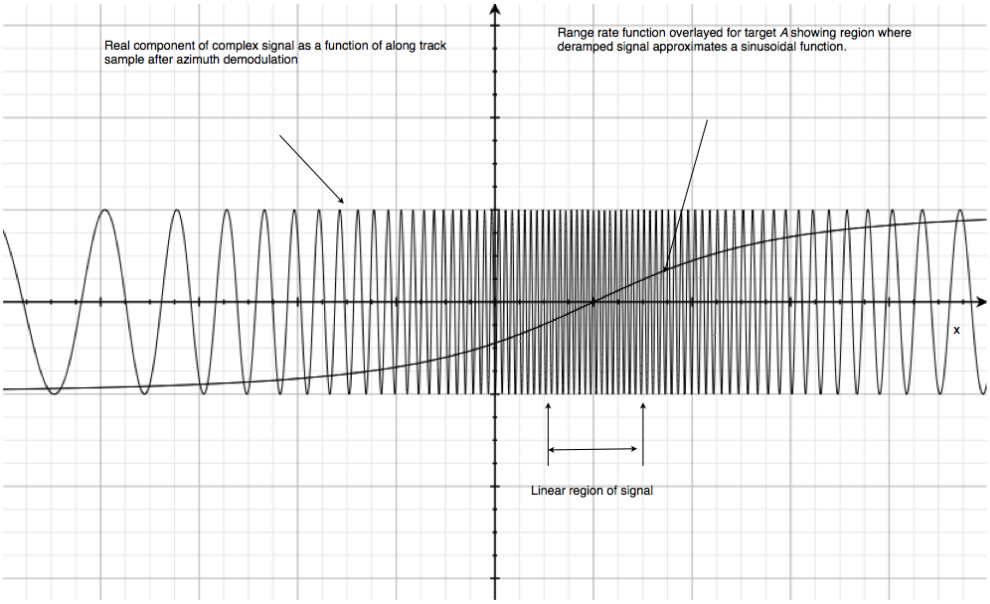


Figure 7.5: De-ramped signal history of Target A after mixing with the conjugate of the expected signal from the scene centre

been de-modulated with the expected signature from the scene centre as described

correctly defined as $\frac{d}{dx}$ rather than $\frac{d}{dt}$

in Chapter 1. The purpose of this process is to leave a constantly progressing phase history from the target where the rate of change of phase is proportional to the target's along-track location from scene centre. In this way the final spotlight image can be formed through a fast Fourier transform (FFT) in the along track direction. The form of Figure 7.5 is therefore provided by multiplying the complex signal by the complex conjugate of the expected signal from scene centre:

$$s(x) = e^{i\frac{4\pi}{\lambda_c}\sqrt{{R_0}^2+(x-X_t)^2}}\, e^{-i\frac{4\pi}{\lambda_c}\sqrt{{R_0}^2+x^2}} \tag{7.4}$$

It should be noted that the form of this equation varies from (1.63) as the deramp-on-receive process has not been included in that derivation. This equations assumes nominal range compression (or fine range resolution) and is omitted so as to clarify the along track effects of the modulating signal.

7.2.1 The Effect of Polar Formatting the Received Signal

It is clear from Figure 7.5 that for small synthetic apertures, Fourier transforming the central region of the signal history after de-modulation would result in an adequate impulse response formation. Unfortunately this is rarely the case for high or even medium resolution SAR systems and so the polar format algorithm (PFA) was developed [14] to interpolate the signal history onto a grid with uniform angle between sensor samples (θ) rather than a uniform spacing (x).

Figure 7.6 sets out the underlying principal behind polar formatting. It was recognised [17] that the most significant cause of defocussing was an inability to adequately correct for range migration or, when the range motion of a sample becomes less than the range resolution of the sensor, to correct the received signal phase history. In order to see why this was an improvement (7.1) is rewritten as a function of θ:

$$R(\theta) = \frac{R_0}{\cos(\theta - \theta_t)} \tag{7.5}$$

Which provides the range as a function of subtended synthetic aperture angle θ where θ_t is the difference in synthetic aperture angle for a given target displaced from the scene centre in the along track direction by X_t. It should be noted that θ_t is

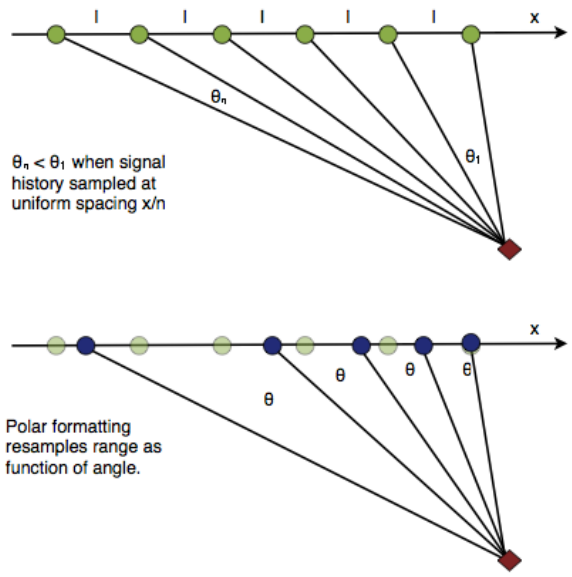


Figure 7.6: Polar reformatting (or polar formatting) resamples the collected signal from equally-spaced along track positions to equally spaced angular positions

itself a function of θ. Assuming that $R_0 >> X_t$:

$$R(\theta) \approx \frac{R_0}{\cos\theta} \tag{7.6}$$

and so the form of the range as a function of aperture angle can be seen in Figure 7.7.

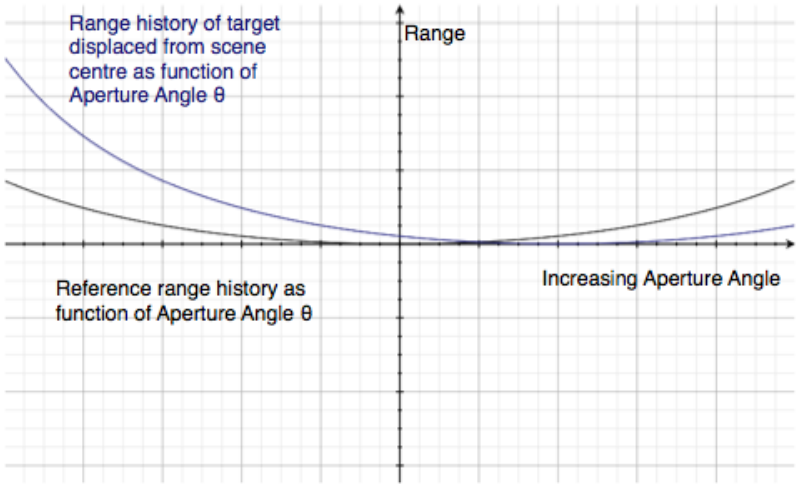


Figure 7.7: Range History of a Target as a Function of Synthetic Aperture Angle θ

Differentiating (7.6) as a function of θ provides :

$$\frac{d}{d\theta}R(\theta) = \frac{R_0 \sin\theta}{\cos^2\theta} \tag{7.7}$$

Which when compared to the original 'rectangular' formatted data (red curve in Figure 7.8) shows that the signal range-rate response (and hence phase-rate or pulse-to-pulse frequency change) now approximates a linear frequency-modulated signal over a much larger region. In essence the form of the phase history signal has been stretched-out by the polar reformatting onto a collection grid that is now non-uniform in time but (approximately) uniform in θ. The final stage of the IFP is to Fourier

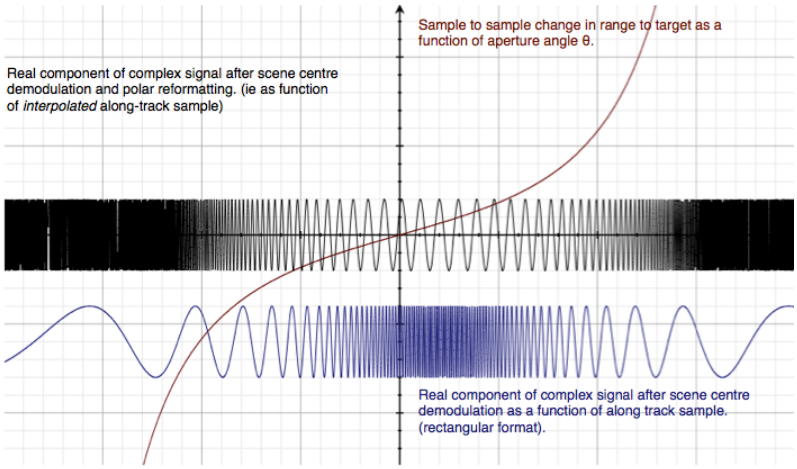


Figure 7.8: Representation of real component of signal from a target within a SAR scene as a function of along track position (Lower blue signal) and synthetic aperture angle θ (central black signal).

transform the resampled data set. Any signals resulting in a constant rate of change of phase within the resampled data set are then collapsed into the impulse response function and of course any signals that have a varying rate of change of phase across the data set will be collapsed into the Fourier waveform representing the range of frequencies within the signal. From this and Figure 7.3 it can therefore be recognised that the polar reformat process ensures that the signal from a radar reflecting target remains at a constant frequency over a larger range of collected samples and hence allows focused imagery to be formed over a longer synthetic aperture dwell time.

7.2.2 Other SAR Image Formation Approaches

Although the analysis to this point has focused on the process of polar reformatting, other techniques such as the Chirp Scaling Algorithm (CSA) [156] [157] and the Range Migration Algorithm (RMA) [21] [22] suffer from the same data support limitations as described here. Namely that the collected radar returns from a target vary non-linearly across the synthetic aperture. Unlike PFA, CSA and RMA do not demodulate their radar returns to scene centre (often called motion compensation to a point), rather they correct for range migration (and hence sample to sample phase changes) by compensating their data to a line parallel to the sensor track. This has the effect of keeping the data as a linear frequency modulated chirp but as can be seen in Figure 7.3, these algorithms still have to resample (or in the case of CSA, phase correct), the data to ensure that the signal history from all reflecting targets maintain a linear, frequency modulated sweep (quadratic rate of change of range / phase) across the synthetic aperture. In this way many scatterers from across the scene can be rendered at the same time significantly saving on the required number of computations. Another approach to SAR processing is the time domain back projection algorithm (TDBPA) [158], [159]. In this technique, each pixel within the final image is formed by searching through the collected signal history for the actual signal returns and then processed independently. Whilst this approach is computationally expensive it does have the advantage that the phase histories can be spatially variant within the scene. Only TDBPA and RMA correct for all signal histories across the entire scene.

7.2.3 Modulating Signatures

Consider a radar reflecting target within a SAR scene that is modulating its range to the platform track at a constant frequency. It is assumed that its RCS is constant over the synthetic aperture. The signature of the target (after required correction of motion through range cells) is given by:

$$s_v(x,t) = \sigma_v e^{i\frac{4\pi}{\lambda_c}R(x)} e^{i\sin(2\pi f_v t)} \tag{7.8}$$

Where σ_v is the vibrating target RCS (assumed constant here) and f_v is the target's frequency of modulation. The phase modulation term at the end of (7.8) gives rise to an additional sinusoidal phase term across the synthetic aperture. After the data is demodulated with the expected signal from the scene centre, the remaining signal will contain the expected constant rate of change of phase term caused by the target's along track location combined with the modulating phase term. Upon image formation via the FFT the expected signature will collapse to a point at the expected location. The modulation term however is given by expansion:

$$\sin(2\pi f_v t) = \frac{e^{i2\pi f_v t} - e^{-i2\pi f_v t}}{2i} \tag{7.9}$$

which has two phase ramps that vary with time, one being the conjugate of the other. As a result the modulating signature is realised after the FFT process as two points (paired echoes).

Its instructive to consider the paired echoes in terms of their supported signal in order to avoid confusion with regard to spatial and temporal modulations. Consider that the FFT process displaces each echo away from its source by one sample per cycle of modulation across the collected data set. A target with exactly one full cycle of modulation within the collected data will be displaced by exactly one sample after being Fourier transformed. If a target modulation contains $N_s/2$ cycles within the N_s samples of data then it will be displaced by $N_s/2$ samples (the maximum Nyquist displacement before ambiguously cycling around the data samples).

The displacement in samples of a vibrating target N_v within a sampled signal is therefore given by

$$N_v = C_v \tag{7.10}$$

With C_v being the number of vibration cycles within the N_s samples of data.

For a SAR system imaging a vibrating target, the modulation frequency is then obtained by dividing the number of cycles of vibration by the collection duration of the data in the synthetic aperture. The vibration frequency is therefore given by:

$$f_v = \frac{N_v}{T_{sa}} \tag{7.11}$$

with T_{sa} being the synthetic aperture duration.

It is common to oversample the radar beam so that the pulse repetition frequency of the radar is larger than the radar beam's illuminated Doppler bandwidth. In addition there may be other oversampling requirements made by the image formation processor prior to final scene focus. This oversampling has the effect of stretching the band-limited signal over more image pixels and so has to be taken into account when determining the displacement of a modulating signal within a SAR image. The final displacement X_v is given by:

$$X_v = N_v O_s = f_v T_{sa} O_s \tag{7.12}$$

and so

$$f_v = \frac{X_v}{O_s T_{sa}} \tag{7.13}$$

where X_v is measured in azimuth image samples and O_s is the azimuth bandwidth oversampling factor. This is a useful expression for vibrating targets in SAR imagery as it emphasizes that the echo displacement is a true-time effect rather than being dependent upon resolution (which is not related to synthetic aperture duration [47]).

Consider now (7.10) and Figure 7.6 and also that polar formatting resamples the collection grid to become a (more) linear function of synthetic aperture angle. In doing so it has the effect of causing the radar returns, collected from points linearly distributed in space (time) to become non-linear. If it is assumed that the platform has a constant velocity across the synthetic aperture then the regularly timed pulse returns will become increasingly stretched out in time as a result of the polar reformat process. For a target modulating at a *constant* frequency this has the inverse effect seen in Figure 7.6 and causes the modulating signature to become stretched out in the collected signal (Figure 7.9). The modulating target now has a range of frequencies

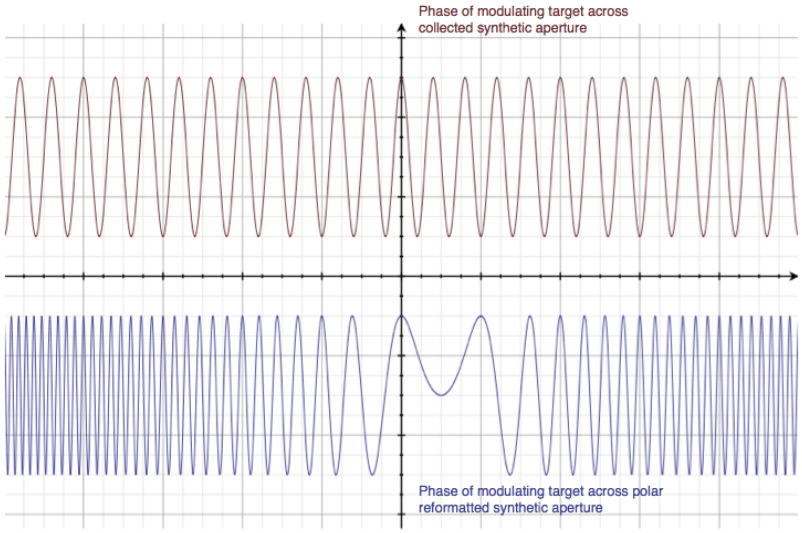


Figure 7.9: Phase of a modulating target as a function of along track position (top red signal) and after polar formatting (bottom blue signal).

after resampling and so after FFT, in the image domain, it now has an along track extent rather than a single impulse response and appears defocused.

7.2.4 The Form of the Modulating Target in the SAR Image

Figure 7.10 provides a cartoon illustration of a modulating point target in a simulated SAR image. The previous section described the azimuth defocus and azimuth displacement of the signature but it is clear that the form of the modulating target also comprises a range defocus term. To understand this effect, consider the holographic nature of SAR Imagery [160]. A high resolution SAR image can be formed by coherently combining several lower resolution 'looks' taken from different sections of the synthetic aperture (Figure 7.11).

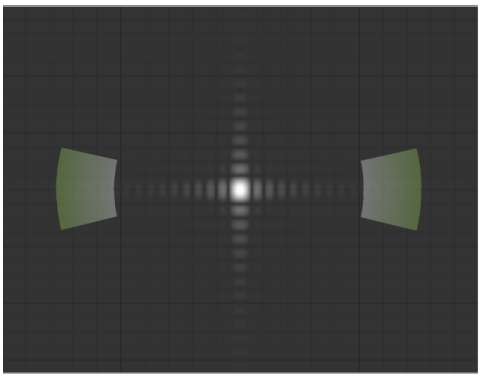


Figure 7.10: Cartoon of a modulating point target and its paired echoes. Range is down

In this image the entire collection aperture has been broken into five sub-apertures or *looks*. Being sub-apertures, the azimuth resolution of each is reduced.

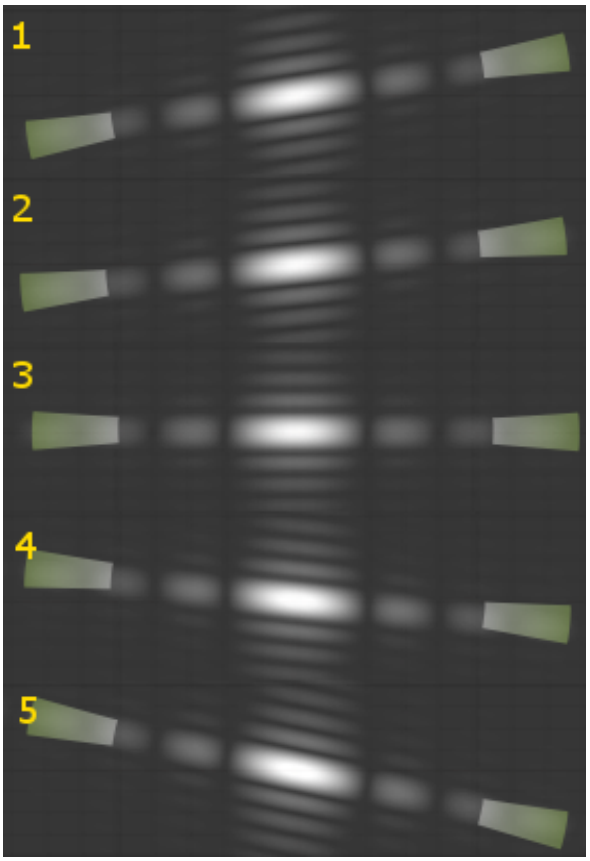


Figure 7.11: Cartoon of a series of 5 contiguous looks of a modulating point target showing how the variation in cross-range direction affects the paired echo signatures

More importantly though the cross-range direction now varies for each look and so the paired echo signatures are no longer displaced horizontally in the image but at an angle commensurate with each sub-aperture's squint angle. The angle of rotation of the paired echoes θ_v around a vibrating target is:

$$\theta_v = \frac{vT_{sa}\left(1 - P_c\right)}{R} \tag{7.14}$$

With v being the velocity of the SAR sensor and P_c the fractional size of each look compared to the full collection. R is the slant range between sensor and target.

The defocussed signature in Figure 7.10 can be thought of as the sum of the cross-range displaced paired echoes in Figure 7.11. It can therefore be seen that although the signatures might sometimes appear to be rectangular (sometime described as 'fuzzballs') they are actually the amalgamation of a series of defocused, azimuth displaced paired echoes that *rotate* around the source of modulation forming

a section of an annulus. The range extent of the signature is caused by the rotation of the sensor across the synthetic aperture and therefore its actual extent is both a function of synthetic aperture angle θ and displacement distance. Of course this form of signature only occurs in the case of a broadside imaging mode and does not accurately match the form of observed signatures which are more rhomboid in nature (Figure 7.1). The final form of modulating signatures is explained by considering the situation when the sensor is not imaging at broadside and by observing the totality of the observations described so far as the synthetic aperture is formed. Consider Figure 7.12.

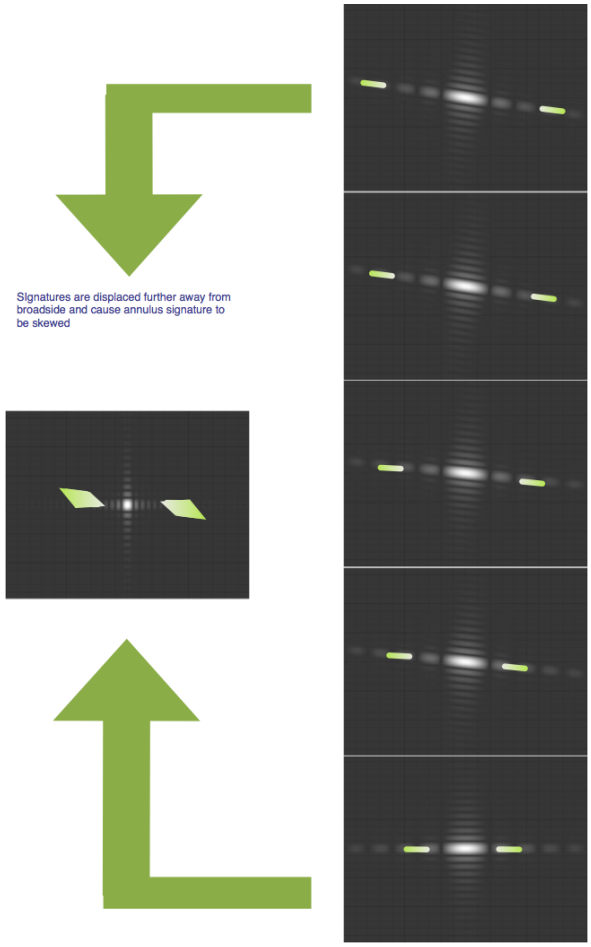


Figure 7.12: Holographic explanation of a modulating target's paired echo formation when the synthetic aperture is collected away from the broadside position.

In this diagram, the sensor is imaging the target by squinting significantly forward and moving from the right to the left. The last multi-look frame is collected

at the broadside position. The target is modulating and so it produces paired echoes that are displaced in the instantaneous cross-range direction. The paired echoes are also defocused due to the requirement to reformat the collection dataset onto a nonuniform-in-time collection grid. From Figure 7.5 and Figure 7.6 it is now obvious that in order that each look maintains the same resolution, the coherent period of integration (CPI) for each look must decrease as the radar moves further towards the broadside point. In this case the first look is further away from broadside and so the duration of the synthetic aperture required to form this look is significantly longer in time than later looks. As the target is modulating with a constant frequency it has more modulation cycles over the collection period for this look and so its paired echo signatures are displaced by a larger number of image samples. On the subsequent looks the coherent period of integration becomes progressively smaller (as the resolution for each look remains the same) and so the paired echoes are displaced by a smaller number of image samples. By coherently combining the looks, the distinctive 'lozenge' shaped signature seen in Figure 7.1 is formed from the resulting 'skewed annulus'.

7.3 Detecting Real-World Vibrating Signatures

Having defined the form of a modulating signal it now remains to determine whether such a signature can be identified in real SAR imagery. As previously discussed, such signatures are likely to be very faint, as the source of modulation has its reflected power distributed into its paired echo signatures. In addition, as described in the previous section, these paired echoes are also each 'defocussed' due to the image formation process. To increase the chances of success, a SAR system called *Bright Spark* was used with a very fine resolution as it provides a high signal-to-clutter ratio thereby increasing the chance of raising a vibration signal above the background and making it detectable.

7.3.1 Bright Spark SAR Data Collection System

The sensor used was a new SAR system called Bright Spark [161], commissioned by Dstl as an experimental platform to explore the art-of-the-possible in fine resolution

imaging and built and maintained by Thales, UK. The sensor parameters for Bright Spark can be see in Table 7.1

Parameter	Specification
Centre Frequency	34.2 GHz
Centre Wavelength	0.00876316 m
Number of Linear FM Chirps	3
Bandwidth of each chirp	1.32 GHz
Chirp step	1.2 GHz
Total bandwidth	3.72GHz
Slant range resolution	0.04 m
Cross range resolution	0.028 m
Image sample size range	0.0375 m
Image sample size cross range	0.0375 m
Range oversample factor	1.33
Cross range Oversample factor	1.33

Table 7.1: Nominal Operating Parameters for Bright Spark SAR System

The project started life as a radar pod for the UK's military drone programme called Watchkeeper [162]. The idea was to push the art of the possible and achieve resolutions beyond conventional SAR systems in order to explore the effects that resolution has on image interpretability and automated feature extraction. Such a system would necessarily demonstrate an order of magnitude increase in data fidelity and collection capability which usually places such payloads into the realms of classified military programmes. As one of Dstl's responsibilities is to facilitate scientific advances it adopted an unusual approach to alleviate this problem. Rather than lock away the Bright Spark data, it worked with decision and policy makers within the British Government to agree that the imagery is not classified unless the area being observed contained sensitive information. For this reason limited Bright Spark datasets have been made available to the scientific community and are therefore able to be used in this project [161].

The radar is quite unusual in that it is centred at 34GHz and achieves its 3.7GHz bandwidth through the use of three 1.3GHz chirps. Each chirp is processed individually, together with any required autofocus solution and then the three images are coherently combined [163].

Figure 7.13 shows one of the initial test images taken over Bruntingthorpe Airfield, UK, together with an optical image of the site for comparison.

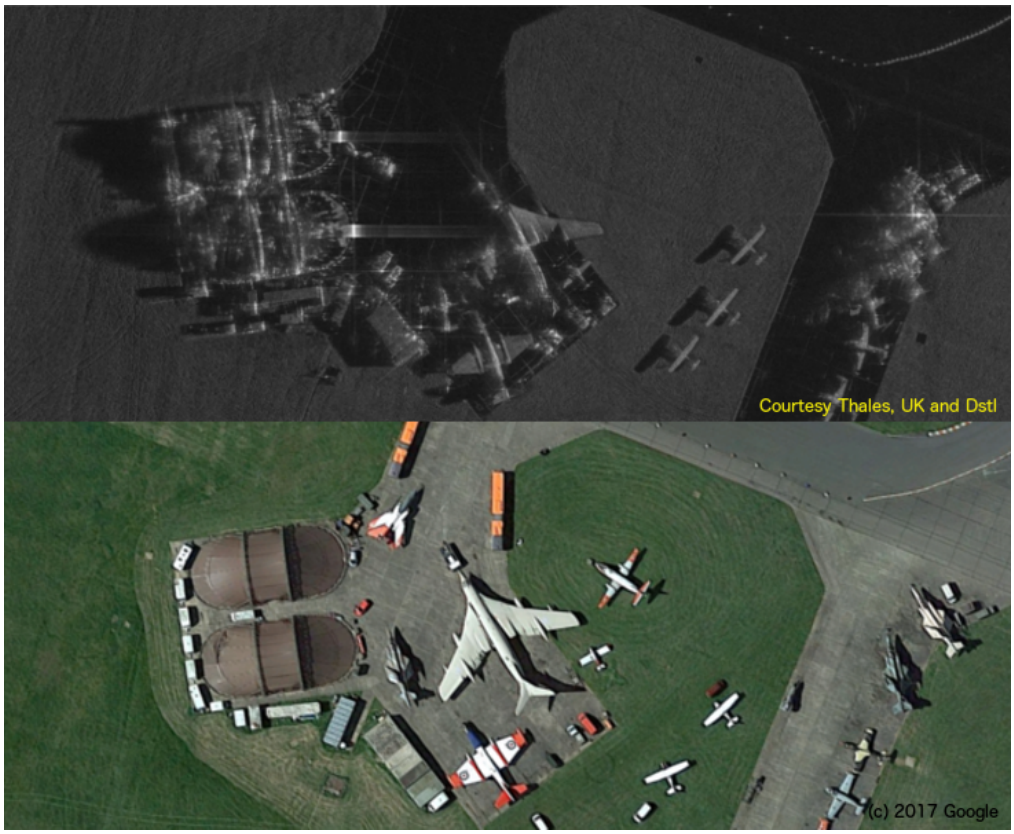


Figure 7.13: Early Bright Spark image (top) taken of Bruntingthorpe Airfield, UK, (52.4961°N, -1.1139°E) with corresponding optical image (taken on a different day)

7.3.2 Vibration Experiment

The system was used in early 2013 to collect imagery from a range of calibrated ground targets in the vicinity of Westcott, Buckinghamshire, UK (51.850472°N, -0.969417°E). This collection contained a generator which was both operating and not operating on subsequent imaging passes. This presented an ideal opportunity to compare the theory to real SAR imagery. An image of the trials scene with inset optical imagery and generator marked can be seen in Figure 7.14

The SAR image of the generator can be seen in the inset of Figure 7.14 and the corresponding ground image can be seen in Figure 7.15. It is interesting to note that the generator has a distinctive brighter point at its far range (bottom left of

Figure 7.14: Brightspark image of Westcott, Buckinghamshire with corresponding optical image (top right). A generator is in the scene (inset). Increasing range is image down.

inset image). The first step was to determine how well SARCASTIC could reproduce the generator. To do this a CAD model was built using Sketchup® (Figure 7.16) and rendered using SARCASTIC and the narrowband data associated with the imaging operation. The results can be seen in Figure 7.17 together with the actual bright spark image for comparison.

The CAD model is clearly not a perfect representation of the actual generator, nevertheless, the simulated image is remarkably accurate with similar spatial features and RCS values. As with the actual SAR image the simulated image also has a bright feature at image bottom left. There is an additional bright feature at image bottom right that is not present in the SAR scene.

Figure 7.15: Image of Generator in Westcott Bright Spark Scene

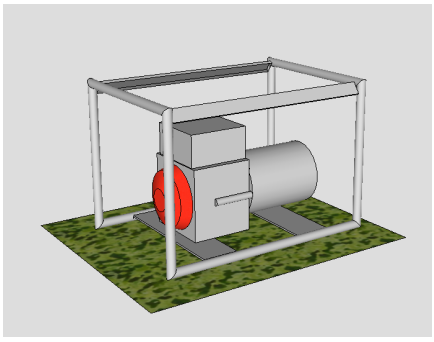


Figure 7.16: Sketchup® Model of Generator

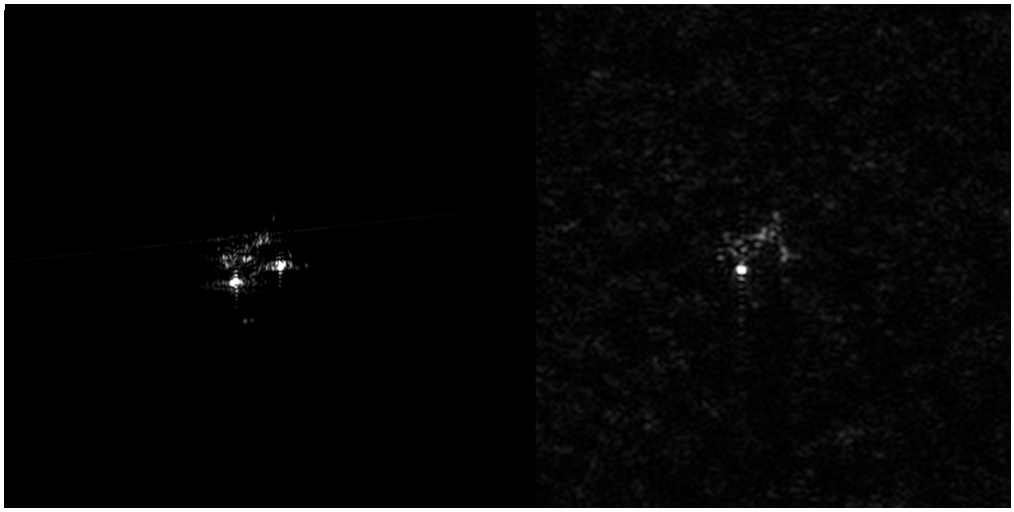


Figure 7.17: Bright Spark image (right) of Generator in Westcott scene together with SARCASTIC simulated image (left) using the same collection parameters and the CAD model in Figure 7.16

As discussed in Chapter 6, any pixel within the formed image can be *'reverse traced'* or *'ray-unravelled'*. This is achieved by identifying the pixel location within the output scene and then determining all the EM field contributions that are associated with that pixel's range (for the provided pulse). Clearly there is a *many-to-one* mapping between the multitudinous paths that a ray can take and the total round trip range and so the different contributions can be ranked in terms of their bounce-number or *order*. Performing this process for the brightest feature on the generator (image bottom left) provides values in table 7.2

It can be seen in this table that the dominant scattering mode has *bounce order 1* or two bounces. In other words the dominant source of the reflections are dihedral

Ray reflection order	Combined EM field (V/m^2)
Bounce order 1	47.289
Bounce order 0	4.234
Bounce order 2	3.776
Bounce order 3	0.0
Bounce order 4	0.0
Bounce order 5	0.0
Bounce order 6	0.0

Table 7.2: Combined field contribution for different bounce modes forming bright feature on generator

in nature. SARCASTIC is able to print out all the rays contributing to this scattering mode which can be seen in Figure 7.18

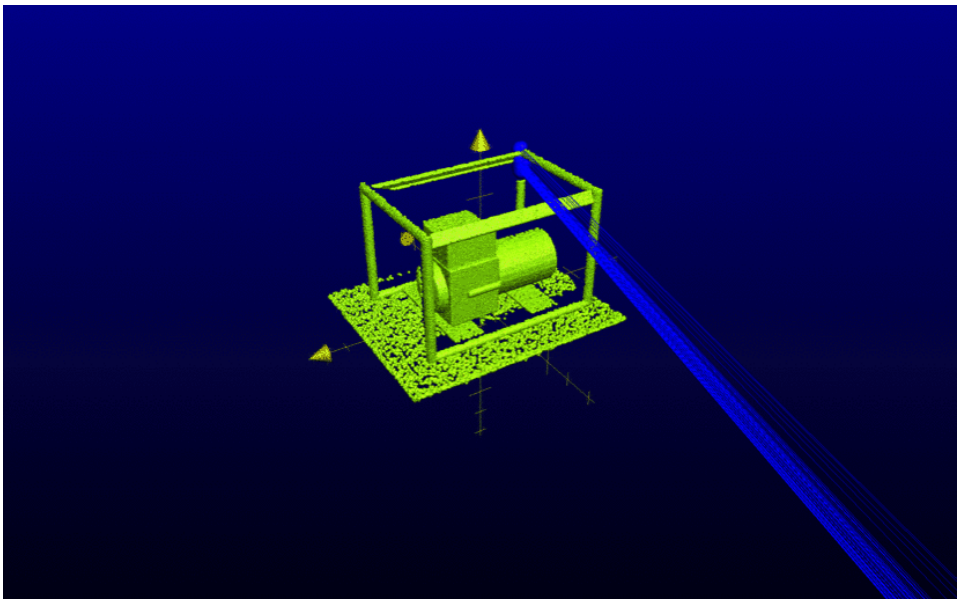


Figure 7.18: Source of dominant scattering modes that contribute to brightest feature in SAR image of generator

In this figure the source of scattering for the bright feature can clearly be identified as a dihedral reflection caused by the 'U-shaped' horizontal frame brace and the frame itself.

Applying the same analysis to the bright feature in the lower right of the SARCASTIC image (that does not appear in actual SAR image) provides a clue to its

[1]This image is actually a movie to better visualize the scattering mechanisms. It was created using the animation package in LaTeX. If it appears as a static image then it may be that your PDF viewer does not support embedded flash. Adobe Reader is freely available from https://get.adobe.com/uk/reader/

source. In this case it is caused by a dihedral reflection from the far left vertical frame post and the ground. Although the SARCASTIC image has attempted to approximate the roughness of the grass, it is obvious from Figure 7.15 and Figure 7.16 that the actual grassy surface is considerably rougher than the CAD model which is why the true image does not contain this feature.

Having determined a CAD model that provides similar results and which has features that are recognisable, it is now possible to set the CAD model of the generator in motion and see how this effects the simulated SAR data.

7.3.3 The SAR Multi-look Movie

The CAD model of the generator can now be set in motion. This is achieved by applying a transformation matrix to the polygonal moving targets before building the k-dimensional tree for the scene. As SARCASTIC is optimised for kd-tree building and traversal efficiency the scene can be repositioned for each pulse with only a slight overhead in rendering performance.

The generator is a 2kVA four-stroke generator providing 50Hz alternating current. Being a four-stroke engine, the power stroke occurs every alternate piston stroke and so a 25Hz vibration frequency was applied to the CAD model and a maximum vibration displacement of 0.0125m was predicted. The results of the SARCASTIC simulation can be seen in Figure 7.19.

Figure 7.19: SARCASTIC simulated SAR image of a 25Hz vibrating 2kVA generator

In this image the paired echoes can be seen as faint signatures displaced in azimuth (left/right). These signatures are similar to those predicted in Section 7.2.4 having both a cross-range and a range component of defocus. As was described

in that section, the instantaneous cross-range direction varies during the image formation process. It is therefore clear that if individual looks were formed of the generator then the paired echoes would be displaced in a different direction in each look. This can be seen in Figure 7.20

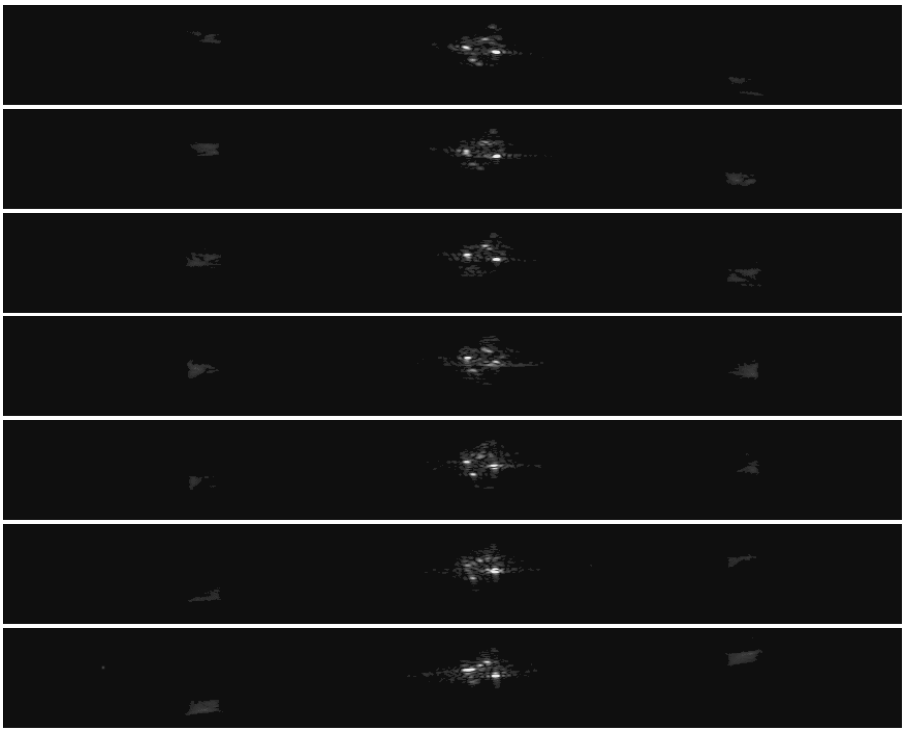


Figure 7.20: Individual *Looks* of the SARCASTIC Simulated Generator

Extending this principal further, each look can be displayed consecutively as a SAR multi look movie (MLM) and the paired echo signatures would be seen to rotate around the source of vibration. This technique has been applied to Figure 7.21 which is an embedded SAR MLM of the vibrating CAD model.

This movie now provides an imagery analyst an important clue of where to look in the SAR image for the tell-tale signatures that the generator is operating. Perhaps more usefully the motion of the signatures is unique only to vibrating/modulating

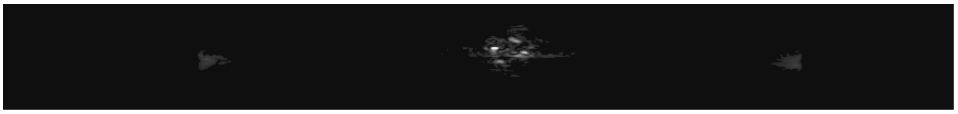


Figure 7.21: SAR Multi-Look Movie (MLM) of SARCASTIC simulation of generator showing paired echoes rotating around the source of vibration[2]

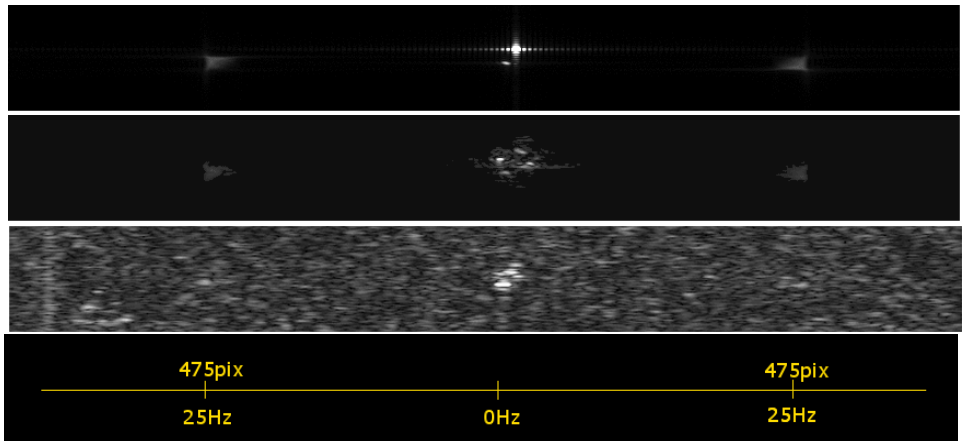


Figure 7.22: Comparison of MLM technique applied to SARCASTIC simulation of point targets (top) and 2kVA generator (2nd down) together with actual Bright Spark image of 2kVA generator (3rd down) and scale (bottom)[3]

targets and, being a moving feature is potentially easier for visual interpretation.

The MLM technique was applied to both the original image and the SARCASTIC model and can be seen in Figure 7.22.

The top image in Figure 7.22 contains an MLM of a simple pair of 10cm sided trihedrals in order to make the vibration signatures more observable. The centre trihedral is static and appears as the bright target in the centre of the image. Towards the bottom left (at the same location as the dihedral on the generator frame) is a second trihedral that is vibrating at 25Hz with an amplitude of 0.0125m.

The second image in Figure 7.22 is the SAR MLM of the SARCASTIC model of the generator. The third image in the sequence contains the MLM of the actual Bright Spark Collection. The bottom image in the sequence provides an image scale in terms of pixels and vibrating frequency.

It can be seen in this image that the SARCASTIC model and the point target simulation provide a useful clue as to where to look in the Bright Spark image. More importantly, the technique reveals the paired echo signatures in the actual SAR image that were undetectable without the process.

To confirm the signatures, the collection parameters are provided in Table 7.3. Application of these figures to equation (7.13) confirms that in this scene the generator is operational and is a 50Hz four-stroke generator vibrating at 25Hz.

Description	Value
Paired echo displacement	475 pixels
Image oversampling factor	1.33
Cycles over synthetic aperture	357.14 modulation cycles
Cross range resolution	0.05 m
Range to image centre	5095.486 m
Synthetic aperture length	893.326 m
Sensor velocity	61.74 m/s
Synthetic aperture duration	14.469 seconds
Vibration frequency	**24.68 Hz**

Table 7.3: Calculation of the Vibration Frequency of the Generator

7.4 Exploration of Unusual Effects

Having shown how vibrating targets can be detected in SAR imagery, it was decided to see if it would be possible to characterise some non-traditional signatures using SARCASTIC as an exploratory tool. For this task a more configurable SAR system was required and so it was decided to use the Cranfield University Ground-Based SAR.

7.4.1 GB-SAR

Figure 7.23: Cranfield University's GB-SAR

The GB-SAR system was designed and built by Cranfield University, UK to provide carefully controlled experimental SAR imagery [77,164,165]. It is a portable stepped frequency continuous wave (CW) imaging radar that can be employed both

indoors and outdoors as seen in Figure 7.23 and, depending on antenna configuration, can produce *hyperspectral*-SAR imaging between 10 and 50GHz [166]. The system consists of a stepper-motor controlled quad-polarisation antenna array mounted on a boom attached to a 'cherry-picker' crane lift. A vector network analyser controls the CW transmitted waveform as it sweeps through the desired frequency range. It also performs the required de-ramping and digitisation prior to image formation.

Image formation is provided using a time-domain back-projection algorithm. An example image of four oil drums (seen in Figure 7.23) can be seen in Figure 7.24.

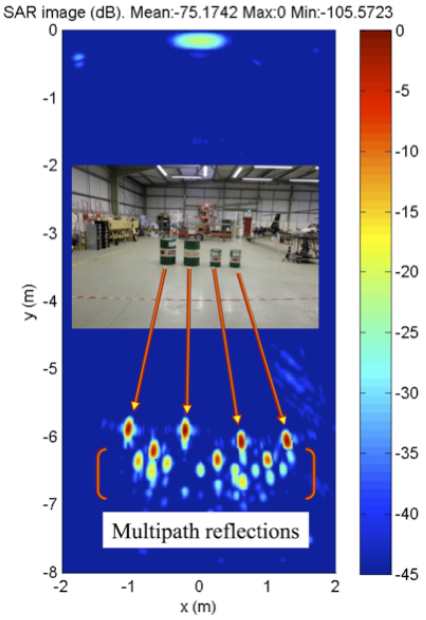


Figure 7.24: An example image from GB-SAR with a photograph of an experimental setup (inset). Dihedral reflections from the 4 oil drums are prominent together with 2nd, 3rd and 4th order multipath reflections between the drums

7.4.2 Multipath Experiments

Before looking at vibration signatures, a comparison of the performance of SARCASTIC in a multipath environment was performed. Figure 7.24 presents quite a challenging scene as it contains four cylindrical steel drums arranged such that their scattered fields interact to create multipath signatures. The accurate simulation of curved surfaces and multipath scattering is particularly challenging for approaches that use ray tracing and PO. The PO approximation requires that facets are planar and so many small facets are required to approximate the curvature of a barrel with the dimensions of each triangle being $\lambda/10$ [167]. This in turn requires many millions of rays to be cast into the scene to ensure that each facet is dealt with appropriately. It should be noted that such a situation is only an approximation for more accurate EM modeling (vs simulation) techniques and the key question is to determine "how good is good enough" for physical understanding of the imaged target?

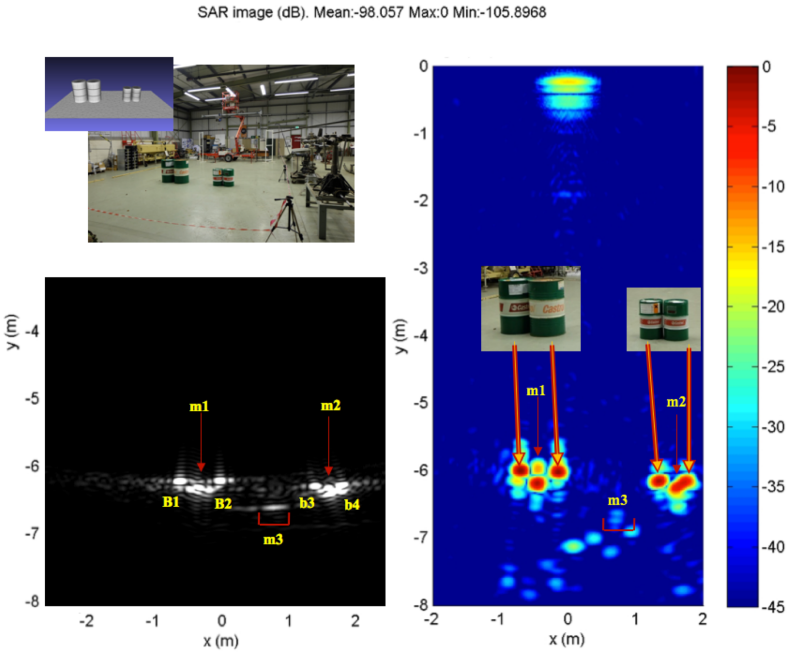


Figure 7.25: Comparison of real and simulated multipath signatures. Top Left: GB-SAR collecting SAR imagery of two large and two small oil drums with CAD model (inset). Right: GB-SAR image of oil drums. SARCASTIC simulation (bottom left) shows the same signatures and facilitates understanding of source of multipath signatures.

The setup and results of the comparison can be seen in Figure 7.25. The scene was setup in the radar facility at Cranfield University and replicated as a CAD model using Sketchup®. Phase history data was then collected using GB-SAR and simulated using SARCASTIC and the resulting data used as input into a back-projection SAR IFP. SARCASTIC was able to predict the virtual multipath scattering between adjacent barrels ($m1$ and $m2$ in the images) in addition to virtual scattering centres between the left pair and right pair of barrels ($m3$ in the images). The source of additional down range scattering signatures within the real GB-SAR data is unknown but may be from more complex features on the oil drums (e.g. a refill lid) that is not present in the CAD model.

7.4.3 Vibration Experiments

To compare vibration results with real data in a controlled environment, an experiment was designed with the Cranfield University GB-SAR and a trihedral mounted to a LEGO Mindstorms® mobile car called a controlled vibration platform (CVP),

Figure 7.26. The CVP is synchronised to move between pulses transmitted and received from GB-SAR and can be programmed to induce a range of vibration signatures.

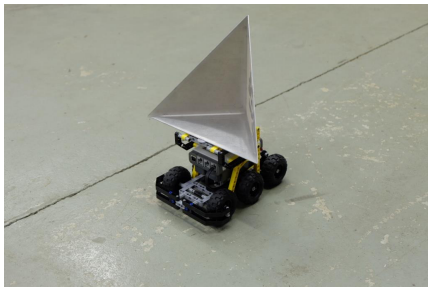


Figure 7.26: Cranfield University's controlled vibration platform (CVP) mounted with a triangular corner reflector. It is built from LEGO Mindstorms® components and synchronised to GB-SAR

An additional capability of both GB-SAR and SARCASTIC is the ability to explore radar penetration phenomenology. As an initial feasibility test and to explore the vibration phenomenology an experiment was performed where the CVP was positioned behind a 9.8*cm* thick, dry concrete block wall and GB-SAR oriented to form a C-band (4.5-6.5GHz) SAR scan whilst the CVP was vibrating. For comparison, a SARCASTIC simulation was also performed. The SARCASTIC results can be seen in Figure 7.27 with the GB-SAR results in Figure 7.28. In both the simulated and real data the

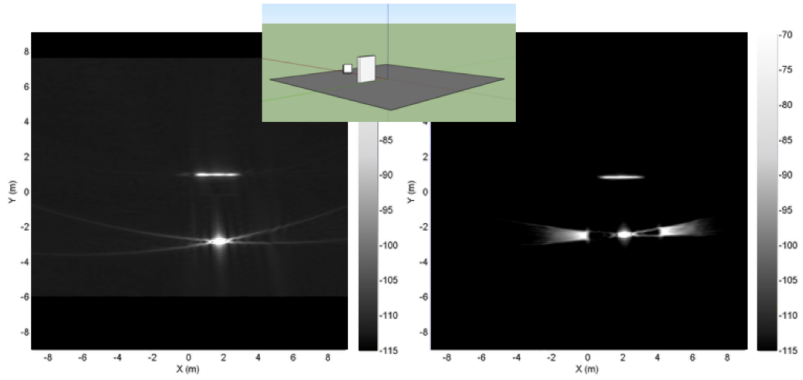


Figure 7.27: SARCASTIC simulation of GB-SAR scene with a 20*cm* square, trihedral positioned behind a static concrete block wall. On the left the trihedral is static and on the right, it is vibrating at a rate of 0.33*Hz* with an amplitude of 5*mm* and an equivalent azimuth resolution of 16*cm*

vibration signatures perform as predicted.

A second experiment was designed to investigate whether the paired echoes could be made to appear asymmetrically about the source of vibration. A set of simple simulations were constructed with SARCASTIC that allowed the vibrating

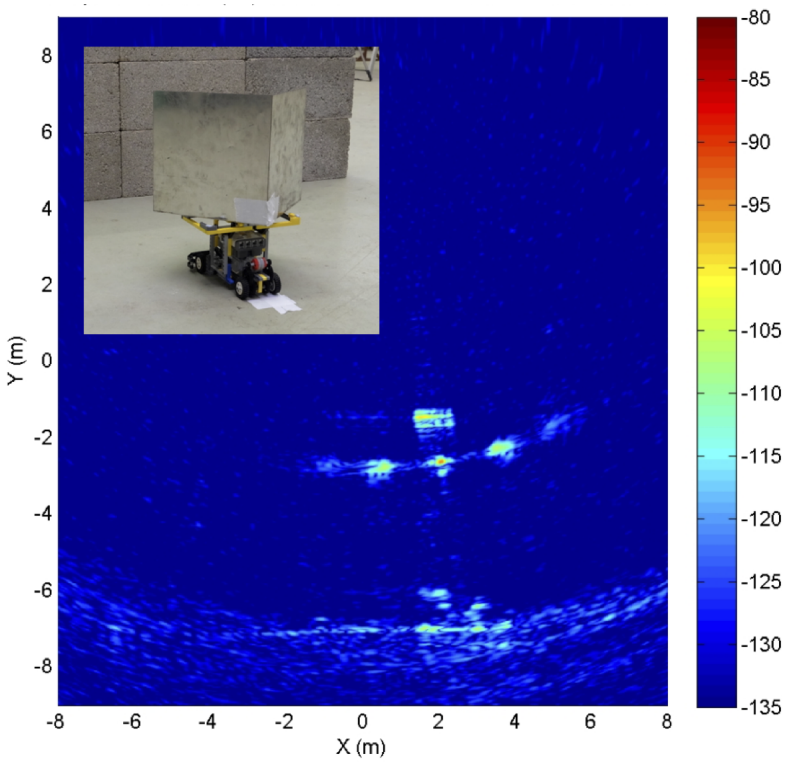


Figure 7.28: GB-SAR image of the CVP vibrating with an amplitude of $5mm$. The scan takes 60 seconds with a pulse separation of $1cm$ and an equivalent vibration frequency of $0.33Hz$). The azimuth resolution is $16cm$

target to move in a sawtooth pattern rather than a sinusoidal motion. (i.e. the CVP would progressively move towards the radar over a range of pulses and then move back to its original position rapidly within a single inter-pulse period). It was postulated that this would represent the effect of reflected energy from the blades of a rotating fan blade. The effect of this type of modulation is to apply an additional frequency component onto the vibrating signal which causes the peak paired echo to be translated in the azimuth direction. The results of the simulation compared to acquired GB-SAR imagery can be seen in Figure 7.29.

The results in Figure 7.29 demonstrate how a sophisticated scattering phenomenology can be decomposed, simulated and then experimentally verified with a true radar system.

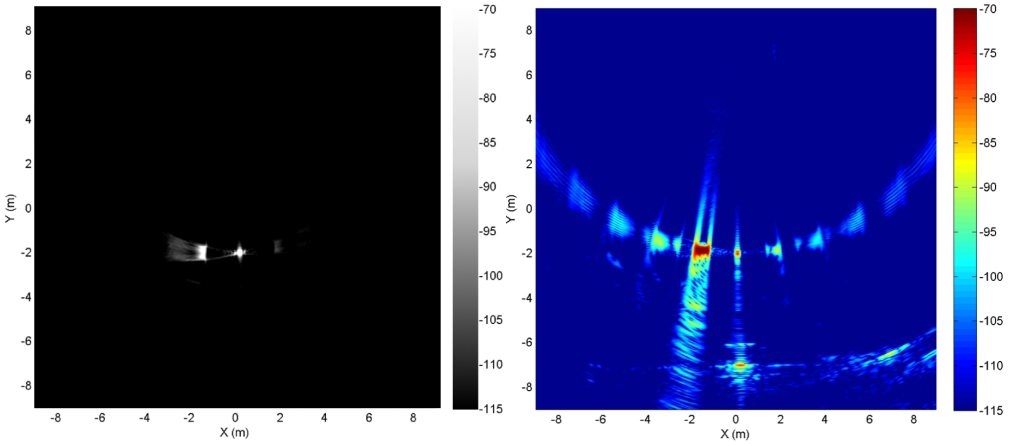


Figure 7.29: A sawtooth vibration applied to a vibrating target. Left is the SARCASTIC simulation. Right is the actual data acquired with GB-SAR. The target vibration frequency is 10Hz with a peak amplitude of 15*mm*.

7.5 Summary

An explanation has been provided for the form of a modulating signal within a SAR collection. It was recognised that a key issue is that SAR imagery is formed independently of time with the key attribute being the rate of change of range from sensor to target. A modulating target however is fixed to the progression of time and so any warping of the time axis required to accurately form SAR imagery has a distinctive and quantifiable effect on the formed imagery. These signatures manifest themselves within the imagery as paired echoes and physical reasons for their structure have been described.

Armed with the knowledge of the form of such targets it is possible to hunt for them in SAR imagery using SARCASTIC as a tool to test hypotheses and to provide clues regarding where to look. This highlighted that the vibration paired echo signatures have a structure caused by the rotation of the axis defining the instantaneous cross-range direction. A technique was provided called the *multi-look movie* that highlights such features in a SAR image.

The approach was verified using the Bright Spark SAR sensor which provided evidence that a generator within a radar scene was operating and was providing a 50Hz AC current.

Finally a series of more sophisticated experiments was performed using the

GB-SAR sensor. These showed that the concept of exploratory hypothesis testing could predict novel signatures that are not usually observed within SAR imagery. In this case a one-sided paired echo signature was predicted. The result was confirmed using the GB-SAR sensor and a novel mobile calibration target operating with a stepped vibration pattern rather than a sinusoidal motion.

Chapter 8

Conclusions

With the growth in SAR sensors, especially in the field of EO, there has been a significant increase in the capabilities of SAR systems in terms of image quality. This has largely made the role of an imagery analyst easier but the specific nature of a SAR image requires a degree of non-literal exploitation skills. Fine resolution SAR imagery contains artifacts that have no optical equivalent, especially if the target is moving or an object was imaged through multiple reflections.

The purpose of this work was to develop a tool that could assist in the exploitation of sophisticated scattering mechanisms. Rather than being purely theoretical the approach developed an electromagnetic ray-tracing tool called SARCASTIC that forms a simulated SAR image with the precise phase-history geometry of a SAR collection requiring analysis.

At the start of this research SARCASTIC had the following requirements:

1. It must provide data in the same format as the phase history from the SAR sensor so that it can be passed through the operational SAR image formation processor. This will ensure that both simulation and actual image have the same processor-induced artifacts.

2. The CAD model must be in an open standard format and be able to be generated by a typical imagery analyst (rather than a computer aided design specialist)

3. It must be able to support hypothesis testing by having a CAD model rapidly modified and a simulated scene produced in less than 15 minutes

4. It must be freely available to scientists and engineers to enhance and improve

5. It must be accessible to imagery analysts for comparison with their standard SAR imagery

This thesis has documented the engineering design of SARCASTIC. First a thorough introduction to electromagnetic field scattering was provided, starting with Maxwell's equations and resulting in a formulation of electric field potential at a receiver when scattered from a planar triangular facet. Particular attention was given to facet material properties and the equations were formulated in terms of electrical resistivity and magnetic conductivity.

The engineering and implementation of SARCASTIC was then discussed in depth, starting with a discussion on validating the integrity of the CAD model in a radar sense (Chapter 3). Then the subtleties of rapid ray-tracing using hierarchical data structures was presented in Chapter 4 followed by the implementation and construction of the capability in Chapter 5.

The tool was then tested and validated against a series of operational SAR sensors. Chapter 6 concentrated on the investigation of multipath effects and used data from DLR's F-SAR system imaging a fire station at DSTL's Porton Down headquarters. It was shown how polarisation can help with understanding of multipath techniques. A technique called ray-unravelling was presented that allows a user to interrogate a feature within a simulated SAR image and identify its likely scattering modes and locations within the building. This approach was used to identify which fire engine bays within the station were occupied (validating previous work performed by DSTL) and also to show that the presence of an unknown object (but **not** a fire engine) was located in one of the bays. This observation was confirmed with ground truth imagery that identified that the source of scattering was a set of storage shelves that would not be directly observable from a stand-off sensor.

During Chapter 4 an approach to fast ray tracing was discussed and it was shown that SARCASTIC is able to match, and in some examples, outperform, computer games in the formulation of hierarchical data structures that help with rapid ray-tracing. This implementation enables SARCASTIC to simulate moving and vibrating targets

within a SAR image. This was the subject of Chapter 7 which compared a SARCASTIC simulation of a vibrating generator with one observed in the Thales/Dstl Bright Spark imagery. A precise understanding of the structure of modulating signatures was presented which enabled a simple technique, called a *multi-look movie* to identify vibrating targets within a SAR scene (providing it has a high signal-to-clutter ratio such as from a fine resolution system). The technique was applied to the generator and it was correctly shown to be vibrating at $25Hz$ which corresponds to a four-stroke engined generator providing $50Hz$ alternating current.

8.1 Summary

In terms of meeting the original objectives stated above, items 1 and 2 have been fully met. Item 3 has been shown to work although the typical run times for SARCASTIC are of the order of 30 minutes for a 100m x 100m scene (or approximately 5 million facets) with scene complexity increasing this number. As a proof of concept this is adequate and, as the software runs on multi-core machines with GPUs and is processing limited, rather than I/O limited, the performance can be easily improved by running across more computers. This would bring the typical run time down to less than 15 minutes.

Item 4 on the list has been successful as the software will be made available on DSTL's internet software repository following completion of the necessary documentation to allow other users to compile, run and enhance the software.

Finally, item 5 has only partially been achieved. This is primarily due to classification challenges regarding the implementation of experimental software on operational systems used by imagery analysts working with sensitive SAR data. The software will however be available to those analysts working at an unclassified level and those scientific staff permitted to use non-accredited sources and methods in their operational support duties.

8.2 Recommendations for Future Work

Applying SARCASTIC to operationally relevant tasks has highlighted several areas where the research may be taken forward. These will now be discussed.

- Perhaps the most obvious recommendation, is the implementation of SARCASTIC on a multi-machine architecture. This would immediately provide performance improvements that may make this approach to SAR image analysis more useful to an exploitation team. This would require the code to be wrapped in a suitable harness that provides node-to-node coordination and queue management such as the Message Passing Interface (MPI). It would allow sophisticated scenes to be rendered using harvested computer power from unused machines within an analytical environment or even using a *'cloud-based'* compute farm.

- If the SAR image formation processor is calibrated and the sensor model within SARCASTIC has the correct effective isotropic radiated power (EIRP), then SARCASTIC will produce calibrated radiometrically correct imagery. This means that features could also be analyzed in terms of their expected radar cross section. This would be useful for assessing radar absorbent materials or objects inside a building and may allow similar shaped objects made from different materials to be discriminated (e.g. for counter-stealth or counter-decoy analysis).

- The phase history data used by SARCASTIC has separate metadata for transmit and receive locations and times. It would be relatively simple to use it to assess the effects of multistatic/bistatic collection strategies on target/feature recognition.

- Being able to rapidly simulate a target vehicle provides the opportunity to generate a large volume of target templates that may be useful to automatic target detection tools and machine learning algorithms.

- A weakness of SARCASTIC is that it only performs a Physical Optics approximation to EM field calculations. It has been assumed that for most large features, such as buildings and vehicles, this is sufficient. Subtle features however such as antennas within a SAR scene are not correctly reconstructed. It would be useful therefore to have the ability to define certain regions within the scene

to have their scattering calculated precisely. An obvious approach would be to define volumes through which finite difference time domain (FDTD) techniques are applied. This would (initially) slow down a processing run but ultimately make for a better tool.

- The formulation of EM scattering equations assumes 'optically flat' facets in order to reconstruct the scattered fields. This means that a curved surface has to be represented by many smaller facets. This causes an inherent error in the calculations caused by the de-coupling between the magnetic and electric fields induced on a surface and results in field strength and surface scattering errors. The application of FDTD techniques discussed above would remove this problem and ultimately permit curved surfaces and electromagnetic metamaterials to be correctly modeled.

Acronyms

A/D	Analogue To Digital
AABB	Axis Aligned Bounding Box
ADC	Analogue To Digital Converter
CAD	Computer Aided Design
CAT	Computer Aided Tomography
CPHD	Common Phase History Data
CPI	Coherent Period Of Integration
CPU	Central Processing Unit
CSA	Chirp Scaling Algorithm
CVP	Controlled Vibration Platform
CW	Continuous Wave
DSTL	Defence Science And Technology Laboratory
EIRP	Effective Isotropic Radiated Power
EM	Electromagnetism Or Electromagnetic
EO	Earth Observation
FDTD	Finite Difference Time Domain
FFT	Fast Fourier Transform
FM	Frequency Modulated

GPGPU	General Purpose Graphical Processing Unit
GPU	Graphical Processing Unit
IF	Intermediate Frequency
IFP	Image Formation Processor
Kd-tree	K Dimensional Tree
MLM	Multi Look Movie
MTRC	Motion Through Range Cells
PEC	Perfect Electrical Conductor
PFA	Polar Format Algorithm
PHD	Phase History Data
PO	Physical Optics
PRF	Pulse Repetition Frequency
Radar	Radio Detection And Ranging
RCS	Radar Cross Section
RF	Radio Frequency
RMA	Range Migration Algorithm
RTRT	Real Time Ray Tracer
RVP	Residual Video Phase
SAH	Surface Area Heuristic
SAR	Synthetic Aperture Radar
SIMD	Single Instruction Multiple Data
SSE	Streaming SIMD Extensions
TBP	Tomographic Back Projection

TDA	Time Domain Algorithm
TDBPA	Time Domain Back Projection Algorithm
TE	Transverse Electric
TEM	Transverse Electromagnetic
TM	Transverse Magnetic
WRRC	Willow Run Research Centre

Glossary

Breadth First

Term to describe the construction of a kd-tree where each level of the tree is considered at the same time with successive loops considering increasing depth levels within the tree.

Bright Spark

The codename for an experimental airborne SAR system funded by Dstl and operated and maintained by Thales, UK.

Chirp

Colloquial term for a linear frequency modulated waveform. So-called because it sweeps up (or down) in frequency like a bird's call.

Common Phase History Data

A format specification for storing synthetic aperture radar data in a phase history data format. The format is designed to be sensor/platform agnostic so that raw SAR data can be shared and processed regardless of any intellectual property or security issues associated with the operation of the sensor.

CUDA

A parallel computing platform and application programming interface created by Nvidia to allow developers to use Nvidia graphics cards as parallel compute engines.

Delaunay Triangulation

A technique for triangulating a mesh such that no vertex point is inside the circumcircle of any triangle.

Depth First

Term to describe the construction of a kd-tree where the entire depth of the tree is constructed for the first node with successive steps returning to parent nodes before attempting to process down the tree as far as possible.

Deramp-on-Receive

A radar technique that down-converts radar returns with those expected from scene centre in order to reduce the required digitisation bandwidth.

GB-SAR

Ground-Based SAR. A radar imaging system developed and operated by Cranfield University, optimised to operate with a large bandwidth.

K-Dimensional Tree

A data structure for organising points in k-dimensional space.

Multilook Movie

A technique to identify unusual scattering phenomenon within a SAR image by breaking the synthetic aperture into individual sub-apertures, that may or may not overlap, and presenting them as an animation or movie.

Multipath

A term used to described delayed returns from a target caused by the propagating waves taking multiple different paths.

OpenCL

Open Computer Language is an open, royalty-free standard for cross-platform parallel computing.

Paired Echoes

Colloquially known as *fuzzballs*. Azimuth displaced signatures from a modulating source in a SAR image.

Phase History Data

Raw pulse data prior to any aperture synthesis. Data can be range compress or range uncompressed.

Physical Optics

The treatment of electromagnetic propagation as complex fields rather than rays.

Polygamous

A term used to describe a triangle that has more than one other triangle attached to any one of its sides.

Posix Threads

Usually referred to as *pthreads*. Provides a way to manage parallel exection of instruction branches within a program operating on a CPU.

Radar

Radio detection and ranging. Detection system using radio waves to detect objects at range.

Ray-Unravelling

A technique to unwind the ray propagation paths within a ray tracing simulator to identify the individual contributions to the final image.

Residual Video Phase

An undesirable phase term present in radar pulses obtained from a radar that operates in a deramp-on-receive mode.

Surface Area Heuristic

A measurement technique for deciding the preferred split plane when splitting a voxel into two child voxels based on the surface area of the parent and child elements and the number of triangular facets that they contain.

Synthetic Aperture Radar

A radar imaging technique that improves the cross-range resolution through coherent combination of pulses.

Triangle Mesh

A data structure of triangles where each triangle side is adjacent to no more than one other triangle side.

Triangle Soup

A data structure of triangles that are largely disorganised with respect to other triangles within the structure.

Warp

A thread group on a GPU. Maximising the occupancy of a warp increases the utilisation or *occupancy* of the device.

Bibliography

[1] A. Love. In memory of Carl A. Wiley. *Antennas and Propagation Society Newsletter, IEEE*, 27(3):17–18, 1985.

[2] C. A. Wiley. Synthetic aperture radars: A paradigm for technology evolution. *IEEE Transactions on Aerospace and Electronic Systems*, AES-21(n. 3):440–443, 1985.

[3] C. W. Sherwin, J. P. Ruina, and R. D. Rawcliffe. Some early developments in synthetic aperture radar systems. *IRE Transactions on Military Electronics*, MIL-6(2):111–115, 1962.

[4] L. J. Cutrona. Synthetic aperture radar. In M. Skolnik, editor, *Radar Handbook*. McGraw-Hill, New York, 1970.

[5] E. N. Leith. A short history of the optics group at the Willow Run laboratories. In Anna Consortini, editor, *Trends in Optical Research, Development and Applications*. Academic Press, San Diego, CA, 1996.

[6] L. J. Cutrona, W. E. Vivian, E. N. Leith, and G. O. Hall. A high-resolution radar combat-surveillance system. *IRE Transactions on Military Electronics*, 1051(2):127–131, 1961.

[7] L. Cutrona, E. Leith, C. Palermo, and L. Porcello. Optical data processing and filtering systems. *IRE Transactions on Information Theory*, 6(3):386–400, 1960.

[8] W. Carrara, R. Goodman, and R. Majewski. *Spotlight Synthetic Aperture Radar Signal Processing Algorithms*, page 143. Artech House, Boston, 1995.

[9] L. J. Cutrona, E. N. Leith, L. J. Porcello, and W. E. Vivian. On the application of coherent optical processing techniques to synthetic-aperture radar. *Proceedings of the IEEE*, 54(8):1026–1032, 1966.

[10] W. M. Brown. Synthetic aperture radar. *IEEE Transactions on Aerospace and Electronic Systems*, AES-3(2):217–229, 1967.

[11] W. M. Brown and L. J. Porcello. An introduction to synthetic-aperture radar. *IEEE Spectrum*, 6(9):52–62, 1969.

[12] B. C. Barber. Theory of digital imaging from orbital synthetic aperture radar. Report 83 079, Royal Aircraft Establishment, 1983.

[13] J. W. Wood, R. G. White, and C. J. Oliver. Distortion free SAR imagery and change detection. In *Proceedings of the 1988 IEEE National Radar Conference*, pages 95–99, Ann Arbor, MI, USA, 1988. IEEE.

[14] J. L. Walker. *Range-Doppler Imaging of Rotating Objects*. Thesis, University of Michigan, 1974.

[15] D. A. Ausherman, A. Kozma, J. L. Walker, H. M. Jones, and E. C. Poggio. Developments in radar imaging. *IEEE Transactions on Aerospace and Electronic Systems*, AES-20(4):363–400, 1984.

[16] J. L. Walker. Range-Doppler imaging of rotating objects. *IEEE Transactions on Aerospace and Electronic Systems*, AES-16(1):23–52, 1980.

[17] J. L. Walker and W. G. Carrara. Method of processing radar data from a rotating scene using a polar recording format, March 4 1980. US Patent 4,191,957.

[18] W. G. Carrara, R. S. Goodman, and R. M. Majewski. *Spotlight Synthetic Aperture Radar: Signal Processing Algorithms*. Artech House, Incorporated, 1995.

[19] D. A. Ausherman and R. S. Powers. Digital image formation processing of spotlight SAR data. *25th Annual Tri-Service RADAR Symposium*, 1979.

[20] C. Cafforio, C. Prati, and F. Rocca. Full resolution focusing of SEASAT SAR images in the frequency-wave number domain. *International Journal of Remote Sensing*, 12(3):491–510, 1991.

[21] F. Rocca, C. Cafforio, and C. Prati. Synthetic aperture radar: A new application for wave equation techniques. *Geophysical prospecting*, 37(7):809–830, 1989.

[22] C. Cafforio, C. Prati, and F. Rocca. SAR data focusing using seismic migration techniques. *IEEE Transactions on Aerospace and Electronic Systems*, 27(2):194–207, 1991.

[23] R. H. Stolt. Migration by fourier transform. *Geophysics*, 43(1):23–48, 1978.

[24] T. Hartley, A. R. Fasih, C. A. Berdanier, F. Ozguner, and U. V. Catalyurek. Investigating the use of GPU-accelerated nodes for SAR image formation. In *IEEE International Conference on Cluster Computing and Workshops 2009.*, pages 1–8. IEEE, 2009.

[25] B. Liu, K. Wang, X. Liu, and W. Yu. An efficient SAR processor based on GPU via CUDA. In *2nd International Congress on Image and Signal Processing*, pages 1–5. IEEE, 2009.

[26] A. Rogan and R. Carande. Improving the fast back projection algorithm through massive parallelizations. In *Radar Sensor Technology XIV*, volume 7669, page 76690I. International Society for Optics and Photonics, 2010.

[27] A. Reigber, M. Jager, A. Dietzsch, R. Hansch, M. Weber, H. Przybyl, and P. Prats. A distributed approach to efficient time-domain SAR processing. In *IEEE International Geoscience and Remote Sensing Symposium*, pages 582–585. IEEE, 2007.

[28] Z. Xiang, K. Wang, X. Liu, and W. Yu. A GPU based time-domain raw signal simulator for interferometric SAR. In *IEEE International Geoscience and Remote Sensing Symposium*, volume 5, pages V–25. IEEE, 2009.

[29] K. A. Graf and H. Guthart. Velocity effects in synthetic apertures. *IEEE Transactions on Antennas and Propagation*, 17(5):541–546, 1969.

[30] R. K. Raney. Synthetic aperture imaging radar and moving targets. *IEEE Transactions on Aerospace and Electronic Systems*, AES-7(3):499–505, 1971.

[31] K. Ouchi. On the multilook images of moving targets by synthetic aperture radars. *IEEE Transactions on Antennas and Propagation*, 33(8):823–827, 1985.

[32] K. D. Ward. Radar imaging of ships at sea. In *IEE Colloquium on Synthetic Aperture Radar*, pages 7–1. IET, 1989.

[33] S. A. S. Werness, W. G. Carrara, L. S. Joyce, and D. B. Franczak. Moving target imaging algorithm for SAR data. *IEEE Transactions on Aerospace and Electronic Systems*, 26(1):57–67, 1990.

[34] K. Ouchi, R. A. Cordey, and R. E. Burge. Statistical analysis of azimuth streaks observed in cassie imagery of ocean surface. In *IEE Colloquium on Interaction of Radiowaves with the Sea Surface*, pages 8–1. IET, 1990.

[35] K. R. Shillington, P. A. Jahans, E. H. Buller, and J. K. E. Tunaley. An ISAR simulator for ships. In *Antennas and Propagation Society International Symposium, 1991. AP-S. Digest*, pages 1032–1035. IEEE, 1991.

[36] N. S. Subotic, L. M. Collins, J. D. Gorman, and B. J. Thelen. A multiresolution approach to target detection in synthetic aperture radar data. In *Conference Record of the Twenty-Eighth Asilomar Conference on Signals, Systems and Computers*, volume 1, pages 122–126. IEEE, 1994.

[37] D. J. Coe and R. G. White. Moving target detection in SAR imagery: experimental results. In *Record of the IEEE 1995 International Radar Conference*, pages 644–649, Arlington, VA, 1995. IEEE.

[38] M. Stuff, M. Biancalana, G. Arnold, and J. Garbarino. Imaging moving objects in 3D from single aperture synthetic aperture radar. In *Proceedings of the IEEE Radar Conference*, pages 94–98, Philadelphia, 2004. IEEE.

[39] L. Wang, L. Zhen, and Y. Guan. Novel space-time processing method for airborne SAR to detect and image the slowly moving targets. In *Third International Joint Conference on Computational Science and Optimization (CSO)*, volume 2, pages 435–439. IEEE, 2010.

[40] Y. Liu, M. Xing, R. Guo, L. Zhang, X. Bai, and Z. Bao. Sandglass transformation for synthetic aperture radar detection and imaging of ship at low signal-to-clutter-plus-noise ratio. *IET Radar, Sonar and Navigation*, 5(3):361–373, 2011.

[41] A. Liu, L. Chen, F. Zhao, and G. Kuang. A novel image based multi-channel SAR-GMTI algorithm. In *International Workshop on Multi-Platform/Multi-Sensor Remote Sensing and Mapping (M2RSM)*, pages 1–4. IEEE, 2011.

[42] R. D. Chapman, C. M. Hawes, and M. E. Nord. Target motion ambiguities in single-aperture synthetic aperture radar. *IEEE Transactions on Aerospace and Electronic Systems*, 46(1):459–468, 2010.

[43] N. S. Subotic, B. J. Thelen, and D. A. Carrara. Cyclostationary signal models for the detection and characterization of vibrating objects in SAR data. In *Conference Record of the Thirty-Second Asilomar Conference on Signals, Systems & Computers*, volume 2, pages 1304–1308. IEEE, 1998.

[44] M. Rüegg, E. Meier, and D. Nüesch. Vibration and rotation in millimeter-wave SAR. *IEEE Transactions on Geoscience and Remote Sensing*, 45(2):293–304, 2007.

[45] M. Pepina and M. M. Hayata. Estimation of vibration spectra including vibrating direction with synthetic aperture radar. In *SPIE Defense, Security, and Sensing*, pages 76990L–76990L. International Society for Optics and Photonics, 2010.

[46] M. Rüegg, E. Meier, and D. Nüesch. Constant motion, acceleration, vibration, and rotation of objects in SAR data. In *Proceedings of SPIE*, volume 5980, pages 598005–1, 2005.

[47] D. Muff, M. Nottingham, D. Andre, B. Barber, D. Blacknell, and H. Griffiths. Characterisation of modulating signatures within sar imagery. In *International Conference on Synthetic Aperture Sonar and Synthetic Aperture Radar*, volume 32, pages 104–109, 2010.

[48] T. Sparr and B. Krane. Time-frequency analysis of vibrating targets in airborne SAR systems. *IEE Proceedings - Radar, Sonar and Navigation*, 150(3):173–176, 2003.

[49] R. Fiedler and R. Jansen. Joint time-frequency analysis of SAR data. In *Proceedings of the Tenth IEEE Workshop on Statistical Signal and Array Processing*, pages 480–484. IEEE, 2000.

[50] K. Ouchi. Modulation of waveheight spectrum and radar cross section by varying surface currents. *IEEE Transactions on Geoscience and Remote Sensing*, 32(5):995–1003, 1994.

[51] K. Ouchi. On the SAR imaging mechanisms of oceanic internal waves. In *International Geoscience and Remote Sensing Symposium*, pages 357–360. IEEE, 1993.

[52] S. Watts, K. D Ward, and R. J. A. Tough. The physics and modeling of discrete spikes in radar sea clutter. In *IEEE International Radar Conference*, pages 72–77, Arlington, VA, 2005. IEEE.

[53] U Soergel, K Schulz, U Thoennessen, and U Stilla. Determination of optimal SAR illumination aspects in build-up areas. In *Proceedings of the IEEE International Geoscience and Remote Sensing Symposium*, volume 6, pages 3662–3664. IEEE, 2003.

[54] Z. Yueting, D. Chibiao, Y. Hongjian, and Q. Xiaolan. The effects of multi-path scattering on the SAR image of cylinder cavity. In *IEEE International Geoscience and Remote Sensing Symposium*, volume 4, pages IV–609. IEEE, 2009.

[55] W. Coburn, C. Kenyon, and C. Le. Preliminary benchmarking of radar signature prediction codes. In *Users Group Conference, 2005*, pages 262–269. IEEE, 2005.

[56] D. B. André. Radar target-ground interaction. In *2007 IET International Conference on Radar Systems*, pages 1–5, Edingburgh, UK, Oct 2007.

[57] V. Krishnan, C. E. Yarman, and B. Yazici. SAR imaging exploiting multi-path. In *IEEE Radar Conference*, pages 1423–1427, Washington, DC, 2010. IEEE.

[58] D. A. Garren, J. S. Goldstein, D. R. Obuchon, R. R. Greene, and J. A. North. SAR image formation algorithm with multipath reflectivity estimation. In *Proceedings of the IEEE International Radar Conference*, pages 323–328, Toulouse,France, 2004. IEEE.

[59] D. A. Garren, D. P. Sullivan, J. A. North, and J. S. Goldstein. Image preconditioning for a SAR image reconstruction algorithm for multipath scattering. In *IEEE International Radar Conference*, pages 300–305, Arlington, VA, 2005. IEEE.

[60] M. Cheney. A mathematical tutorial on synthetic aperture radar. *SIAM review*, 43(2):301–312, 2001.

[61] A. Ribalta. Time-domain reconstruction algorithms for FMCW-SAR. *IEEE Geoscience and Remote Sensing Letters*, 8(3):396–400, 2011.

[62] S. V. Tsynkov. On the use of start-stop approximation for spaceborne SAR imaging. *SIAM Journal on Imaging Sciences*, 2(2):646–669, 2009.

[63] D. Deirmendjian. *Electromagnetic Scattering on Spherical Poly-Dispersions*. Elsevier, New York, 1969.

[64] G. T. Ruck. *Radar Cross Section Handbook*, volume 1. Plenum Press, the University of Michigan, 1970.

[65] J. Scheer, J.A. Scheer, and J.L. Kurtz. *Coherent Radar System Performance Estimation*. Artech House, Incorporated, 1993.

[66] S. R. J. Axelsson. Frequency and azimuthal variations of radar cross section and their influence upon low-frequency SAR imaging. *IEEE transactions on geoscience and remote sensing*, 33(5):1258–1265, 1995.

[67] A. Danklmayer, B. J. Doring, M. Schwerdt, and M. Chandra. Assessment of atmospheric propagation effects in SAR images. *IEEE Transactions on Geoscience and Remote Sensing*, 47(10):3507–3518, 2009.

[68] D. C. Livingston. *The Physics of Microwave Propagation*. Prentice-Hall, 1970.

[69] Z-W. Xu, J. Wu, and Z-S. Wu. A survey of ionospheric effects on space-based radar. *Waves in Random Media*, 14(2):S189–S274, 2004.

[70] F. J. Meyer and J. B. Nicoll. Prediction, detection, and correction of faraday rotation in full-polarimetric L-band SAR data. *IEEE Transactions on Geoscience and Remote Sensing*, 46(10):3076–3086, 2008.

[71] K. Davies. *Ionospheric Radio*. IET, 1990.

[72] ITURP Recommendation. 838-3. specific attenuation model for rain for use in prediction methods. *ITU-R Recommendations, P Series Fasicle, ITU, Geneva, Switzerland*, 2005.

[73] M. P. M. Hall. Effects of the troposphere on radio communication. *IEE Electromagnetic Waves Series*, 8, 1980.

[74] A. W. Doerry. Atmospheric loss considerations for synthetic aperture radar design and operation. In *Radar Sensor Technology VIII and Passive Millimeter-Wave Imaging Technology VII*, volume 5410, pages 17–28. International Society for Optics and Photonics, 2004.

[75] H. Nyquist. Certain topics in telegraph transmission theory. *Transactions of the American Institute of Electrical Engineers*, 47(2):617–644, 1928.

[76] H. D. Griffiths and C. J. Baker. *Radar Imaging for Combatting Terrorism*, pages 29–48. Springer, 2007.

[77] D. André, K. Morrison, D. Blacknell, D. Muff, M. Nottingham, and C. Stevenson. Very high resolution coherent change detection. In *IEEE International Radar Conference*, pages 0634–0639, Johannesburg, South Africa, 2015. IEEE.

[78] C. A. Balanis. *Antenna Theory: Analysis and Design*. Wiley, 2012.

[79] C. A. Balanis. *Advanced Engineering Electromagnetics*. Wiley, 2012.

[80] L. Tsang, J.A. Kong, and K.H. Ding. *Scattering of Electromagnetic Waves, Theories and Applications*. Wiley, 2000.

[81] D. C. Jenn. *Radar and Laser Cross Section Engineering*. American Institute of Aeronautics and Astronautics, 2005.

[82] E. F. Knott, J. Shaeffer, and M. Tuley. *Radar Cross Section, Second Edition*. Institution of Engineering and Technology, 2004.

[83] S. Silver. *Microwave Antenna Theory and Design*. P. Peregrinus, 1949.

[84] J. B. Schneider. Understanding the finite-difference time-domain method. *School of electrical engineering and computer science Washington State University.*, 2010. http://www.eecs.wsu.edu/~schneidj/ufdtd/, Accessed: 29.11.2012).

[85] R. P. Feynman, R. B. Leighton, and M. L. Sands. *The Feynman Lectures on Physics*. Pearson/Addison-Wesley, 1963.

[86] N. I. Faros. Radar cross section synthesis for planar resistive surfaces. Report, DTIC Document, 1994.

[87] T. B. A. Senior. Combined resistive and conductive sheets. *IEEE transactions on antennas and propagation*, 33(5):577–579, 1985.

[88] T. B. A. Senior. Impedance boundary conditions for imperfectly conducting surfaces. *Applied Scientific Research, Section B*, 8(1):418–436, 1960.

[89] F. Weinmann. Ray tracing with PO/PTD for RCS modeling of large complex objects. *IEEE transactions on antennas and propagation*, 54(6):1797–1806, 2006.

[90] A. Ludwig. Computation of radiation patterns involving numerical double integration. *IEEE Transactions on Antennas and Propagation*, 16(6):767–769, 1968.

[91] R. J. Pogorzelski. The Ludwig integration algorithm for triangular subregions. *Proceedings of the IEEE*, 73(4):837–838, 1985.

[92] M. L. X. Dos Santos and N. R. Rabelo. On the ludwig integration algorithm for triangular subregions. *Proceedings of the IEEE*, 74(10):1455–1456, 1986.

[93] F. J. S. Moreira and Jr. Prata, A. A self-checking predictor-corrector algorithm for efficient evaluation of reflector antenna radiation integrals. *IEEE Transactions on Antennas and Propagation*, 42(2):246–254, 1994.

[94] E. E. Garrido. *Graphical User Interface for a Physical Optics Radar Cross Section Prediction Code*. Phd thesis, Naval Postgraduate School Monterey, CA, Storming Media, 2000. http://books.google.com/books?id=9doaAAAACAAJ.

[95] F. Chatzigeorgiadis. *Development of Code for a Physical Optics Radar Cross Section Prediction and Analysis Application*. Phd thesis, Naval Postgraduate School Monterey, CA, Storming Media, 2004. http://books.google.com/books?id=3EuWAAAACAAJ.

[96] Trimble. SketchUp. https://www.sketchup.com, 2018. Accessed 5th January 2018.

[97] Autodesk. Maya. https://www.autodesk.co.uk/products/maya/overview, 2018. Accessed 5th January 2018.

[98] Sketchup. 3d warehouse. https://3dwarehouse.sketchup.com/index.html, 2018. Accessed 5th Jan 2018.

[99] The collada document object model. https://www.khronos.org/collada/wiki/COLLADA_DOM, 2008. Accessed: 24 Nov 2017.

[100] P. Bourke. Ply - polygon file format. http://paulbourke.net/dataformats/ply/, 2014. Accessed 18th Dec 2017.

[101] B. N. Delone. Sur la sphère vide. a la mémoire de george voronoi. *Izv. AN OMEN*, 1934.

[102] The CGAL Project. *CGAL User and Reference Manual*. CGAL Editorial Board, 4.11 edition, 2017.

[103] R. Shewchuk, J. Triangle: Engineering a 2D quality mesh generator and delaunay triangulator. In *Applied Computational Geometry: Towards Geometric Engineering*, volume 1148, pages 203–222. Springer-Verlag, May 1996.

[104] A. M. Helmenstine. Table of electrical resistivity and conductivity. `http://chemistry.about.com/od/moleculescompounds/a/Table-Of-Electrical-Resistivity-And-Conductivity.htm`, July 2017. Accessed: 2017-Nov-24.

[105] J. M. Waddell. Scattering from rough plates. Report, DTIC Document, 1995.

[106] T. Möller and B. Trumbore. Fast, minimum storage ray triangle intersection. *Journal of Graphics Tools*, 2(1):21–28, 1997.

[107] D. Badouel. *An Efficient Ray Polygon Intersection*. Graphics Gems III. Academic Press ISBN: 0124096735, 1992.

[108] A. Woo. *Fast Ray-Polygon Intersection.* Graphics Gems. Academic Press p394, 1990.

[109] J. Erickson. Pluecker coordinates. *Ray Tracing News*, 10(3):11, 1997.

[110] I. Wald. Realtime ray tracing and interactive global illumination. *Ausgezeichnete Informatikdissertationen 2004*, 2005.

[111] Agner F. Instruction tables. `http://www.agner.org/optimize/instruction_tables.pdf`, May 2017. Accessed 2nd Dec 2017.

[112] A. Glassner. *An Introduction to Ray Tracing*. Morgan Kaufmann. ISBN: 0-12286-160-4, 1989.

[113] S. M. Rubin and T. Whitted. A three-dimensional representation for fast rendering of complex scenes. *Computer Graphics*, 14(3):110–116, 1980.

[114] T. L. Kay and J. T. Kajiya. Ray tracing complex scenes. *Computer Graphics (Proceedings of SIGGRAPH 86)*, 20(4):269–278, 1986.

[115] B. Sproull and W. M. Newman. *Principles of Interactive Computer Graphics*. McGraw-Hill Education, International edition, ISBN: 0-07-085535-8, pp124-252, 1973.

[116] Wikipedia. Cohen-Sutherland algorithm. `https://en.wikipedia.org/wiki/Cohen-Sutherland_algorithm`, 1967. Accessed 18th Dec 2017.

[117] J. Amanatides, A. Woo, et al. A fast voxel traversal algorithm for ray tracing. In *Eurographics*, volume 87(3), pages 3–10, 1987.

[118] K. Klimaszewski and T. W. Sederberg. Faster ray tracing using adaptive grids. *Computer Graphics and Applications, IEEE*, 17(1):42–51, 1997.

[119] M. Gigante. Accelerated ray tracing using non-uniform grids. *Proceedings of Ausgraph '90*, pages 157–163, 1988.

[120] P-K. Hsiung and R. Thibadeau. Accelerating ARTS. *The Visual Computer*, 8(3):181–190, 1992.

[121] A. Fujimoto, T. Tanaka, and K. Iwata. ARTS: Accelerated ray-tracing system. *Computer Graphics and Applications, IEEE*, 6(4):16–26, 1986.

[122] D. Cohen. Voxel traversal along a 3d line. *Graphics gems IV*, 4:366, 1994.

[123] A. S. Glassner. Space subdivision for fast ray tracing. In I. Joy Kenneth, W. Grant Charles, L. Max Nelson, and Hatfield Lansing, editors, *Tutorial: Computer Graphics; Image Synthesis*, pages 160–167. Computer Science Press, Inc., 1988.

[124] H. Samet. Implementing ray tracing with octrees and neighbor finding. *Computers & Graphics*, 13(4):445–460, 1989.

[125] K-Y. Whang, J-W. Song, J-W. Chang, J-Y. Kim, W-S. Cho, C-M. Park, and I-Y. Song. Octree-R: An adaptive octree for efficient ray tracing. *IEEE Transactions on Visualization and Computer Graphics*, 1(4):343–349, 1995.

[126] E. Haines. Efficiency improvements for hierarchy traversal in ray tracing. In James Arvo, editor, *Graphics Gems II*. Academic Press, San Diego pp 267-272, 1991.

[127] B. Smits. Efficiency issues for ray tracing. *J. Graph. Tools*, 3(2):1–14, 1998.

[128] V. Havran. Heuristic ray shooting algorithms. *PhD Thesis, Faculty of Electrical Engineering*, Czech Technical Univesrity in Prague, 2001.

[129] K. R. Subramanian and D. S. Fussell. Factors affecting performance of ray tracing hierarchies. Report, University of Texas at Austin, 1990.

[130] K. Sung and P. Shirley. Ray tracing with the bsp tree. In Kirk David, editor, *Graphics Gems III*, pages 271–274. Academic Press Professional, Inc., 1992.

[131] J. Arvo and D. Kirk. Fast ray tracing by ray classification. *SIGGRAPH Comput. Graph.*, 21(4):55–64, 1987.

[132] George Simiakakis. *Accelerating Ray Tracing With Directional Subdivision and Parallel Processing*. Thesis, University of East Anglia, 1995.

[133] S. Popov, J. Günther, H-P. Seidel, and P. Slusallek. Stackless kd-tree traversal for high performance GPU ray tracing. In *Computer Graphics Forum*, volume 26, pages 415–424. Wiley Online Library, 2007.

[134] Vlastimil Havran. Statistical Comparison of Ray-Shooting Efficiency Schemes. Report, Department of Computer Science, 2000.

[135] I. Wald and V. Havran. On building fast kd-trees for ray tracing, and on doing that in O (N log N). In *IEEE Symposium on Interactive Ray Tracing*, pages 61–69. IEEE, 2006.

[136] David J. MacDonald and Kellogg S. Booth. Heuristics for ray tracing using space subdivision. *Vis. Comput.*, 6(3):153–166, 1990.

[137] J. Bittner. Hierarchical techniques for visibility computations. *Prague: Department of Computer Science and Engineering, Czech Technical University*, 2002.

[138] V. Havran and J. Bittner. On improving kd-trees for ray shooting. *Journal of WSCG*, 10:209–216, 2002.

[139] Henry Fuchs, Zvi M. Kedem, and Bruce F. Naylor. On visible surface generation by a priori tree structures. *SIGGRAPH Comput. Graph.*, 14(3):124–133, 1980.

[140] Dan Gordon and Shuhong Chen. Front-to-back display of bsp trees. *IEEE Comput. Graph. Appl.*, 11(5):79–85, 1991.

[141] Frédo Durand. *3D Visibility: Analytical Study and Applications*. PhD thesis, PhD thesis, Universite Joseph Fourier, Grenoble, France, 1999.

[142] T. Möller and E. Haines. *Real-Time Rendering*. A K Peters, Ltd, 1999.

[143] M. R. Kaplan. Space-tracing: A constant time ray-tracer. *SIGGRAPH85 Tutorial on the Uses of Spatial Coherence Ray Tracing*, 1985.

[144] Luis Santalo. *Integral Geometry and Geometric probability*. Cambridge University Press, ISBN 0521523443, 2002.

[145] L. Szécsi. An effective implementation of the kd-tree. In Jeff Lander, editor, *Graphics Programming Methods*, pages 315–326. Charles River Media, 2003.

[146] M. Pharr and G. Humphreys. *Physically Based Rendering: From Theory to Implementation*. Morgan Kaufman, 2004.

[147] K. Zhou, Q. Hou, R. Wang, and B. Guo. Real-time kd-tree construction on graphics hardware. *ACM Trans. Graph.*, 27(5):1–11, 2008.

[148] S. Sengupta, M. Harris, Y. Zhang, and J. D. Owens. Scan primitives for GPU computing. In *Graphics hardware*, volume 2007, pages 97–106, 2007.

[149] Stanford University. The stanford 3d scanning repository. `http://graphics.stanford.edu/data/3Dscanrep/`, 2014. Accessed 18th Dec 2017.

[150] Wikipedia. The unix philosophy. `https://en.wikipedia.org/wiki/Unix_philosophy`, 2017. Accessed 5th Jan 2018.

[151] National Geospatial-Intelligence Agency. Compensated phase history data (CPHD). Technical Report NGA.STAN.0068-1, National Geospatial-Intelligence Agency, 2016.

[152] K.E. Dungan, L.A. Gorham, and L.J. Moore. SAR digital spotlight implementation in matlab. In *Algorithms for Synthetic Aperture Radar Imagery XX*, volume 8746, page 87460A. International Society for Optics and Photonics, 2013.

[153] ESA. Polarimetry tutorial, 2018. Accessed 26th January 2018.

[154] D. Blacknell, Andre D., J. Potter-Hall, J. Brown, and C. Stevenson. Dual Polarimetric SAR Exploitation. Technical Report CR72989/1.0, DSTL, 31 May 2013.

[155] V. C. Chen and R. Lipps. Advanced Synthetic Aperture Radar Imaging and Feature Analysis. Report, DTIC Document, 2003.

[156] R. K. Raney. A phase preserving SAR processor. In *International Geoscience and Remote Sensing Symposium*, volume 4, pages 2588–2591, Vancouver, 1989.

[157] H. Runge and R. Bamler. A novel high precision SAR focussing algorithm based on chirp scaling. In *International Geoscience and Remote Sensing Symposium*, volume 1, pages 372–375, Houston, 1992.

[158] M. D. Desai and W. K. Jenkins. Convolution backprojection image reconstruction for spotlight mode synthetic aperture radar. *IEEE Transactions on Image Processing*, 1(4):505–517, 1992.

[159] M. Backues, U. Majumder, D. York, and M. Minardi. Synthetic aperture radar backprojection on Sony Playstation 3 cell broadband engine and Intel quad-core Xeon. *High Performance Embedded Computing (HPEC)*, 2008.

[160] C. V. Jakowatz, D. E. Wahl, P. H. Eichel, D. C. Ghiglia, and P. A. Thompson. *Spotlight-Mode Synthetic Aperture Radar: A Signal Processing Approach*, volume 101. Kluwer Academic Publishers Boston, USA:, 1996.

[161] D. G. Muff. Bright spark synthetic aperture radar imagery. `https://dstl.github.io/brightspark-data/`, 2017, (accessed 1st October 2017).

[162] Wikipedia: Thales watchkeeper wk450. `https://en.wikipedia.org/wiki/Thales_Watchkeeper_WK450`, 2014, (accessed 1st October 2017).

[163] M. Stevens, O. Jones, P. Moyse, S. Tu, and A. Wilshire. Bright spark: Ka-band SAR technology demonstrator. In *IET Radar Conference*, Belfast, UK, 2017.

[164] K. Morrison, J. C. Bennett, G. Cookmartin, A. J. McDonald, A. Race, and S. Quegan. Three-dimensional X-band SAR imaging of a small conifer tree. *International Journal of Remote Sensing*, 22(4):705–710, 2001.

[165] S. C. M. Brown, S. Quegan, K. Morrison, J. C. Bennett, and G. Cookmartin. High-resolution measurements of scattering in wheat canopies-implications for crop parameter retrieval. *IEEE Transactions on Geoscience and Remote Sensing*, 41(7):1602–1610, 2003.

[166] K. Morrison, D. André, D. Blacknell, D. Muff, M. Nottingham, C. Stevenson, and J. Bennett. Hyperspectral 10-50GHz SAR imaging of building materials. In *Proceedings of the International Radar Conference*, pages 1–5, Guangzhou, China, 2016. IEEE.

[167] A. K. Öztürk. *Implementation of Physical Theory of Diffraction for Radar Cross Section Calculations*. PhD thesis, Bilkent University, 2002.

Colophon

This document was set in the Times Roman typeface using LaTeX and BibTeX. It was written using *Sublime Text 3*, primarily on a 2014 Mac Pro[1] but also a mid-2012 MacBook Pro Retina[1]. Illustrations were developed using *GIMP*, and *Keynote*[1]

[1]Mac Pro®, MacBook Pro® and Keynote® are trademarks of Apple Inc